

LAW  
B-9  
SAR

B/c →







Giovanni SARTOR

LA RAPPRESENTAZIONE DELLA CONOSCENZA  
NELLE APPLICAZIONI GIURIDICHE  
DELL'INTELLIGENZA ARTIFICIALE

Tesi presentata al fine del conseguimento del  
titolo di Dottore dell'Istituto Universitario Europeo  
alla Commissione giudicatrice

Commissione di tesi: Prof. J.L. GARDIES (Université de Nantes)  
Prof. P. NERHOT (Istituto Universitario Europeo)  
Prof. E. PATTARO (Università di Bologna)  
Prof. G. TEUBNER (Istituto Universitario Europeo)  
Prof. J. WROBLEWSKI (University of Lodz)

Firenze, novembre 1989





(21)

R0880

LAW B89

SAR



GIOVANNI SARTOR

**LA RAPPRESENTAZIONE DELLA CONOSCENZA  
NELLE APPLICAZIONI GIURIDICHE  
DELL' INTELLIGENZA ARTIFICIALE**



## INTRODUZIONE

Il presente volume ha per oggetto la rappresentazione formale di contenuti giuridici al fine dell'introduzione di tecnologie informatiche nel diritto. In particolare, la formalizzazione della conoscenza rappresenta una condizione necessaria dello sviluppo di applicazioni giuridiche dell'intelligenza artificiale.

Si presenteranno metodi e problemi emersi nelle ricerche e nelle realizzazioni operative, facendo particolare riferimento ad un'esperienza in corso presso l'Università di Bologna, il progetto IRI<sup>1</sup>.

I problemi da affrontare nello sviluppo di sistemi informatico-giuridici "intelligenti" sono di varia natura: informatici, logici, giuridici, filosofici, l'approccio deve essere necessariamente interdisciplinare. Pertanto, se si dedicherà particolare attenzione agli aspetti filosofico-giuridici, non ci si sottrarrà all'esigenza di trattare taluni aspetti tecnici. Per ciascuna delle principali tematiche trattate, si farà precedere all'approfondimento della considerazione delle applicazioni giuridiche, una succinta introduzione al contesto scientifico e tecnologico in cui queste si collocano.

Oggi una prima fase dell'informatizzazione del mondo del diritto, quella che ricorreva a metodi informatici tradizionali - e che ha condotto, in particolare, alla realizzazione dei sistemi di documentazione giuridica automatica - può dirsi compiuta, almeno per quanto riguarda i problemi teorici. Vi sono numerosi problemi ancora irrisolti, soprattutto per quanto attiene all'efficienza degli strumenti informatici, e alla definizione di contesti organizzativi che ne consentano un utile impiego, ma prospettive e limiti di queste tecnologie sembrano delinearsi con precisione. Quegli strumenti possono incidere in misura limitata sull'attività del giurista, modificando il contesto in cui quell'attività si svolge, piuttosto che il lavoro giuridico in senso stretto.

Da alcuni anni, l'interesse della ricerca si è spostato verso una nuova disciplina, la c.d. *intelligenza artificiale*, che si propone di sviluppare sistemi informatici che possano simulare comportamenti intelligenti. Si parla sempre più frequentemente di sistemi esperti giuridici, cioè di sistemi capaci di svolgere prestazioni analoghe a quelle di un giurista esperto; vengono presentati nel dibattito scientifico, e talvolta anche sul mercato,

---

<sup>1</sup>Il progetto IRI (Irnerio Regioni Italiane), si propone di realizzare applicazioni informatiche nel campo del diritto dell'ambiente e di sviluppare un "guscio" per sistemi basati sulla conoscenza giuridica.

programmi informatici che si qualificano come "esperti" o come aspiranti tali.

In realtà non esiste nessuna applicazione giuridica delle tecniche dell'intelligenza artificiale, che abbia mostrato una apprezzabile utilità nella vita giuridica. Tuttavia, guardando al di là delle realizzazioni già compiute, non sembra vi siano ostacoli insuperabili ad un utile impiego dei metodi dell'intelligenza artificiale nel diritto<sup>2</sup>. Sembra dubbio, almeno nei limiti degli sviluppi tecnologici oggi prevedibili, che possa mai realizzarsi un vero sistema esperto giuridico, cioè un programma informatico in grado di esibire una competenza simile a quella di un giurista esperto. Piuttosto che di sistemi esperti preferisco parlare di sistemi basati sulla conoscenza, sistemi automatici capaci di realizzare un uso flessibile della conoscenza per risolvere determinati tipi di problemi.

Nel mondo del diritto vi sono interessanti prospettive per l'uso di sistemi basati sulla conoscenza. Tuttavia, la loro realizzazione pone numerosi problemi ai quali non si è ancora trovata una soluzione soddisfacente, allo stato della tecnologia informatica e degli studi logici e filosofico-giuridici. Ancora più numerosi sono i problemi per i quali non vi è una soluzione universalmente accettata, che richiedono scelte tra le diversi approcci alternativi. Si tratta non solo delle scelte da effettuare nello sviluppo di un sistema esperto giuridico, ma anche di quelle relative alla determinazione dei settori del diritto e dei contesti istituzionali nei quali l'uso di sistemi siffatti può rivelarsi opportuno. Tali scelte richiedono valutazioni ispirate da considerazioni relative alle funzioni sociali del diritto, della sua interpretazione e applicazione, scelte - in un senso ampio - politico-giuridiche. Solo grazie al riconoscimento della complessità e l'ampiezza di queste scelte, e quindi delle responsabilità che ineriscono alla progettazione e all'uso di sistemi di conoscenza nel diritto, sarà possibile indirizzare correttamente lo sviluppo di queste nuove tecnologie.

Sarebbe grave che tali scelte sfuggissero al controllo e al dibattito critico dei giuristi, i soli che possano apprezzare la rispondenza delle soluzioni tecnologiche alle funzioni sociali appena menzionate. Sarebbe d'altro canto un errore se gli stessi giuristi ritenessero che il loro compito possa risolversi nel fornire ad un sistema automatico una conoscenza, una competenza che il sistema possa poi usare autonomamente, senza un continuo controllo.

---

<sup>2</sup>Quei metodi hanno già trovato utili applicazioni in numerosi settori, come la medicina, la chimica, la biologia, l'economia.



Infatti, l'intelligenza automatica trova precisi limiti, indipendenti, almeno in una certa misura, dalle attuali potenzialità tecnologiche. Un sistema dotato di intelligenza giuridica artificiale, piuttosto che come un giurista automatico dotato di competenze analoghe a quelle di un giurista umano, può essere visto come un preciso esecutore, cui siano state affidate indicazioni dettagliate (è questa la conoscenza da impartire al sistema automatico), con il compito di dedurre le conseguenze per casi futuri. Chi ha impartito tali indicazioni al sistema, e, comunque chi dispone del suo impiego, non può sottrarsi alla responsabilità per il suo operato: di qui la necessità di un penetrante controllo sulle premesse da cui muove il sistema automatico, le sue regole di inferenza, le conclusioni cui giunge, la sua adeguatezza nei diversi contesti di utilizzo.

Un analogo pericolo di disconoscimento delle proprie responsabilità si verifica per il teorico del diritto, che è chiamato a dare un contributo fondamentale nello sviluppo di sistemi basati sulla conoscenza giuridica. Si è proposto di risolvere una volta per tutte il problema della base teorica delle applicazioni giuridiche dell'intelligenza artificiale, individuando il punto d'incontro tra i diversi approcci filosofico-giuridici<sup>3</sup>, un insieme di teorie condivise da tutte le principali correnti della teoria del diritto e sufficienti a fondare lo sviluppo di sistemi esperti. Anche i teorici del diritto, dopo aver fornito la propria conoscenza avrebbero esaurito il proprio contributo. Contro questa prospettiva semplificatrice, è facile osservare che, anche se un consenso delle diverse teorie del diritto esistesse, esso potrebbe fondare solo soluzioni banali e prive non solo di interesse teorico, ma anche di utilità pratica. Per quanto riguarda i principali problemi filosofico-giuridici che le applicazioni di intelligenza artificiale sollevano non vi sono premesse che non siano discutibili, e talvolta nessun approccio consente una soluzione del tutto soddisfacente (si pensi, ad esempio, al problema della logica deontica, o a quello del ragionamento analogico).

Come vedremo nel seguito, la caratteristica peculiare dei sistemi informatici intelligenti è la loro capacità di usare in modo flessibile grandi quantità di informazione, di trasformare dati in conoscenza. Pertanto, il problema fondamentale, nella realizzazione di questi sistemi, attiene alla rappresentazione della conoscenza: la conoscenza necessaria per risolvere i problemi di cui il sistema dovrà occuparsi va formulata in strutture formali che possano essere manipolate da un programma informatico. La

---

<sup>3</sup>Questo approccio è sviluppato da SUSSKIND R.E., *Expert systems in Law*, Claredon Press, Oxford, 1988.

formalizzazione della conoscenza giuridica è particolarmente difficile, poiché il linguaggio giuridico, benché parzialmente tecnicizzato, è immerso nel linguaggio naturale e ne conserva tutta la complessità.

L'impiego di un linguaggio formale sufficientemente ricco da consentire un'agevole rappresentazione della conoscenza giuridica, ma suscettibile di un'efficiente trattamento computazionale, rappresenta la condizione necessaria per lo sviluppo di applicazioni giuridiche dell'intelligenza artificiale. La discussione del problema della rappresentazione della conoscenza giuridica consentirà di individuare le prospettive e i limiti dell'intelligenza artificiale nel mondo del diritto, i modi nel quale queste nuove tecnologie informatiche possono essere indirizzate e rese compatibili con le funzioni sociali del diritto, le scelte e i contributi che il giurista è chiamato ad operare in questo nuovo contesto tecnologico.

Il presente lavoro si divide in 2 parti:

(a) La prima parte è un introduzione alla problematica delle applicazioni giuridiche dell'intelligenza artificiale. Dapprima si ripercorrerà brevemente l'evoluzione dell'informatica (cap. 1) e dell'informatica giuridica (cap. 2). Quindi, ci si soffermerà più ampiamente sui sistemi di conoscenza (cap. 3), e sui sistemi di conoscenza giuridica (cap. 4).

(b) Nella seconda si approfondirà il tema della rappresentazione della conoscenza. Si illustrerà il problema della formalizzazione (l'uso di linguaggi formali) per lo sviluppo di sistemi informatici (cap. 5) e, in particolare, di sistemi informatici giuridici (cap. 6). Si esamineranno poi alcuni dei metodi e dei linguaggi usati nell'intelligenza artificiale per la rappresentazione della conoscenza (7) e le loro applicazioni giuridiche (cap. 8).

L'esame delle applicazioni e dei linguaggi dell'intelligenza artificiale nel diritto consentirà di ricondurre questa disciplina al modello o paradigma della logica formale, e, in particolare, del sistema assiomatico deduttivo. Con riferimento a questo paradigma, cercheremo di riconsiderare limiti e prospettive dell'uso di sistemi informatico-giuridici intelligenti (cap. 9).

Il volume è completato da un'appendice nella quale si descrive il sistema IRINORM, da me sviluppato nell'ambito del summenzionato progetto IRI.

Questo contributo è destinato soprattutto a persone di formazione giuridica, interessate ad una introduzione in una problematica ancora nuova, ma forse destinata ad avere presto un impatto sul loro lavoro. Spero che possa rivelarsi utile anche a persone di formazioni informatica, per le quali le applicazioni di intelligenza artificiale nel diritto possono presentare un notevole interesse, non solo

sotto il profilo applicativo, ma anche per le importanti questioni generali che sollevano.



## PARTE I.

# INFORMATICA E AUTOMAZIONE DEL LAVORO INTELLETTUALE.

L'affermazione che l'informatica sta penetrando in tutti i settori delle società moderne, che la sua diffusione - la c.d. informatizzazione - rappresenta un tratto caratteristico della società moderna, è ormai diventato un luogo comune. La convivenza con macchine che manipolano informazioni è una necessità per tutti e gli strumenti informatici fanno parte dell'ambiente sociale, tanto che la loro presenza a volte passa inavvertita. L'attenzione per le moderne tecnologie dell'intelligenza artificiale, non deve far dimenticare che esse rappresentano solo l'ultima tappa di un più ampio fenomeno, al quale vanno ricondotte.

Così, anche le prospettive delle applicazioni giuridiche dell'intelligenza artificiale vanno considerate nell'ambito della progressiva occupazione del mondo del diritto da parte dell'informatica, nell'ambito dell'informatizzazione del diritto.

Nel primo capitolo, si considererà il processo dell'informatizzazione. Si ricostruiranno le tappe dell'occupazione della vita sociale da parte dell'informatica, in ciascuna delle quali si cercherà di individuare un paradigma scientifico e tecnologico determinato.

Nel secondo, si passerà ad esaminare il corrispondente processo nel mondo del diritto. Si considereranno le realizzazioni e le prospettive delle diverse fasi dell'informatizzazione di contesti giuridici.

Nel terzo, ci si soffermerà sui sistemi di conoscenza, la tecnologia caratteristica dell'intelligenza artificiale, illustrandone i tratti essenziali, nel loro sviluppo storico.

Infine, nel quarto ed ultimo capitolo, si considereranno i sistemi di conoscenza giuridici. Si ricorderemo brevemente le principali realizzazioni e progetti in corso, e si cercherà di delineare le possibili applicazioni di questa tecnologia nel mondo del diritto.





## CAP. 1

# L'AUTOMAZIONE DEL LAVORO INTELLETTUALE: DALLA CALCOLATRICE ELETTRONICA AI SISTEMI DI CONOSCENZA

Come si è spesso affermato, le tecnologie informatiche automatizzano, "macchinizzano" il processo di elaborazione delle informazioni, il lavoro intellettuale. Le tecnologie precedenti avevano realizzato l'automazione del lavoro manuale. L'informatica realizza una sorta di "seconda rivoluzione industriale": essa estende l'industrializzazione, cioè la sostituzione di macchine al lavoro umano, anche alle attività umane intellettuali. L'uso di tecnologie informatiche nei diversi contesti sociali - la c.d. informatizzazione - ha pertanto un profondo impatto sociale, che comprende sia aspetti comuni a tutte le forme di industrializzazione che aspetti specifici, caratteristici dell'automazione del lavoro intellettuale<sup>1</sup>.

La progressiva estensione dell'informatica a nuovi settori della vita sociale e a nuovi tipi di attività, è in stretta correlazione con il progresso scientifico e tecnologico: le tecnologie informatiche rendono possibile l'automazione di nuovi ambiti del lavoro intellettuale, e l'ampliamento e l'accelerazione della diffusione dell'informatica costituiscono, a loro volta, lo stimolo del progresso scientifico e tecnologico.

E' dato distinguere tre fondamentali fasi dell'informatizzazione, in ciascuna delle quali si realizza l'occupazione tecnologica di un'area di lavoro intellettuale. A ciascuna di esse corrisponde un determinato tipo di tecnologie informatiche, caratterizzato da strutture idonee a svolgere una particolare specie di attività intellettuale. Tali strutture individuano un determinato

---

<sup>1</sup>Cfr. STEINMÜLLER W., *Informationstechnologie Folgen Informationstechnologie und Informationsrecht III*, in "Nachrichten für Dokumentation", 1981, pp. 241-255, il quale distingue, tra gli effetti sociali dell'informatica, (a) gli effetti di industrializzazione, quali aumento di produttività, disoccupazione (e contemporanea creazione di nuovi posti di lavoro), riorganizzazione, socializzazione e taylorizzazione del lavoro, effetti che derivano, seppure in diversa misura, da tutte le forme di industrializzazione e (b) gli effetti di informatizzazione, tra i quali, in particolare, il passaggio a "strutture e regolarità industriali nella ripartizione sociale del potere" dipendente dall'informazione (e quindi lo sviluppo di nuove forme di controllo sociale e il turbamento degli equilibri di potere). Gli effetti di informatizzazione si riconnettono specificamente con la "macchinizzazione" del lavoro intellettuale.

modello o paradigma dell'informatica<sup>2</sup>. Le tre fasi in questione sono: calcolo numerico, gestione dei dati, intelligenza artificiale.

Come è noto, un elaboratore elettronico può svolgere solo compiti che siano riducibili ad un algoritmo: un algoritmo è una ricetta (una sequenza di prescrizioni o "istruzioni") che indica in modo preciso e non ambiguo i passi che debbono essere compiuti per risolvere correttamente, a partire da determinate informazioni, un certo tipo di problema (se una soluzione esiste), in un tempo finito<sup>3</sup>. Per poter essere eseguiti da un elaboratore,

---

<sup>2</sup>Cfr. KUHN T., *La struttura della rivoluzioni scientifiche*, Einaudi, Torino, 1969 (traduzione di *The Structure of Scientific Revolution*, Chicago, University of Chicago Press, 1962), il quale usa il termine *paradigma* per "far presente il fatto che alcuni esempi di effettiva prassi scientifica riconosciuti come validi - esempi che comprendono globalmente leggi, teorie, applicazioni e strumenti - forniscono modelli che danno origine a particolari tradizioni di ricerca scientifica con una loro coerenza" (pp. 29-30). Secondo lo stesso autore, lo sviluppo della scienza procede per successive rivoluzioni scientifiche caratterizzate da un cambiamento di paradigmi. Nel nostro contesto questi concetti non debbono essere intesi in un senso eccessivamente rigido: ciascuna delle fasi che individueremo è caratterizzata da un diverso modello di informatica, che acquista un ruolo predominante, ma nelle aree nelle quali si sono rivelati idonei, rimangono spesso metodi e tecniche sviluppati in precedenza. Inoltre, la ricostruzione di diversi paradigmi nell'evoluzione di una scienza può dipendere dal particolare interesse che motiva l'indagine storica. Qui, siamo interessati a trovare un collegamento tra le tecnologie e il loro impatto sociale. Per una diversa ricostruzione, cfr. LANZARONE G.A., *Informatica*, in *Gli strumenti del sapere contemporaneo, Volume I, Le Discipline, Grande dizionario enciclopedico*, Utet, 1985, p. 391 ss., il quale distingue nello sviluppo dell'informatica tre fasi, dominate rispettivamente dai paradigmi di ricerca empirico, matematico e ingegneristico (cfr. WEGNER P., *Research Paradigms in Computer Software*, in *Proceedings of the International Conference on Reliable Software*, Los Angeles 21, 32 aprile 1975, 1975). Lanzarone individua nell'informatica "capace di riconciliare il linguaggio del pensiero con quello l'azione, necessitando solo di materie prime di larga disponibilità ... il nuovo paradigma (scientifico e culturale) della società dell'informazione" (LANZARONE G.A., *Informatica*, cit., p. 399).

<sup>3</sup>KNUTH T.S., *The Art of Computer Programming*, Addison Wesley, Reading (Mass), 1973 (2<sup>a</sup> ediz.), pp. 7-8, definisce l'algoritmo come "un insieme finito di regole che da una sequenza di operazioni per risolvere uno specifico problema", insieme dotato delle seguenti caratteristiche: (a) finitezza (l'algoritmo termina in un numero finito di passi), (b) definitezza (ogni passo è esattamente specificato), (c) input (quantità assegnate prima dell'inizio dell'algoritmo), (d) output (quantità che risultano dall'algoritmo e che hanno una relazione determinata con gli ingressi), (e) effettività (ogni passo è sufficientemente elementare da poter essere direttamente eseguito dal sistema).

gli algoritmi vanno formulati in appositi linguaggi formali, i c.d. linguaggi di programmazione. I moderni elaboratori sono tutti riconducibili ad un unico modello, la c.d. macchina di von Neumann<sup>4</sup>. La macchina di von Neumann si compone fondamentalmente di una grande memoria ripartita in celle, nelle quali possono essere registrati dei dati, e di un'unità centrale di elaborazione, dotata di alcune celle locali di memoria, chiamate registri. L'elaborazione dei dati si svolge come segue: i dati registrati nella memoria vengono trasferiti nei registri, l'unità centrale usa i dati memorizzati nei registri per compiere operazioni aritmetiche e confronti, i risultati delle operazioni vengono trasferite dai registri alla memoria. Un programma per una macchina di von Neumann consiste, pertanto, nella descrizione/prescrizione di una sequenza di operazioni elementari sui dati, combinate con istruzioni di controllo, che modificano l'ordine nel quale vengono eseguite le operazioni sui dati.

Nelle tre fasi dell'informatizzazione possiamo rilevare una modificazione del concetto di algoritmo (di programma)<sup>5</sup>, l'adozione di metodi e di linguaggi sempre più astratti e meno dipendenti dal modello tecnologico cui si ispirano gli attuali elaboratori.

## 1. Le operazioni di calcolo numerico

La prima fase dell'informatizzazione è stata caratterizzata dall'uso dell'informatica soprattutto per effettuare operazioni di calcolo numerico. Gli elaboratori elettronici, come già le calcolatrici meccaniche furono destinati soprattutto all'effettuazione di calcoli complessi<sup>6</sup>.

Per eseguire operazioni di calcolo l'elaboratore (che allora veniva chiamato prevalentemente "calcolatore" o computer) doveva essere dotato fondamentalmente della possibilità di effettuare operazioni aritmetiche e confronti tra simboli, operazioni che potessero essere combinate in una sequenza definita con precisione: a questi fini poteva

---

<sup>4</sup>Questo modello fu sviluppato da J. Von Neumann negli anni quaranta.

<sup>5</sup>Si suole distinguere tra l'algoritmo (il metodo astratto per risolvere un certo problema) e il programma (la formulazione dell'algoritmo in un determinato linguaggio di programmazione). Qui non saranno comparati diversi linguaggi di programmazione rispetto al medesimo algoritmo e, pertanto, possiamo usare i due termini come sinonimi.

<sup>6</sup>Sembra che le prime operazioni affidate agli elaboratori siano stati i calcoli delle traiettorie dei proiettili. Cfr. LOSANO M., *Informatica per le scienze sociali*, Einaudi, Torino, p. 141 ss.

bastare un algoritmo concepito nei termini delle operazioni della macchina di von Neumann.

La teoria degli algoritmi, che "si occupa dell'esistenza di algoritmi effettivi per calcolare determinate quantità"<sup>7</sup> ha costituito la base teorica che ha consentito la prima fase dell'informatizzazione, fase caratterizzata appunto dal paradigma dell'algoritmo.

## 2. La gestione di dati

La seconda fase dell'informatizzazione è consistita nell'estensione dell'automazione alla gestione o elaborazione di dati più in generale (data processing). Ci si è accorti che il "calcolatore" poteva servire a memorizzare grandi masse di dati, anche non numerici. Tali masse di dati potevano fungere da supporto per elaborazioni algoritmiche.

Si aprivano nuove, enormi, possibilità di espansione per l'informatica, come la gestione e l'uso di archivi, basi di dati, sistemi documentari, la memorizzazione e l'elaborazione di testi.

Per affrontare questi nuovi compiti sono stati necessari non solo nuovi supporti materiali, come le memorie di massa sempre più capaci e veloci (nastri e soprattutto dischi con enormi capacità di memorizzazione), ma anche un nuovo paradigma, un nuovo apparato concettuale: la programmazione strutturata. Le operazioni elementari sui dati (assegnazione di valori a variabili, operazioni aritmetiche, confronti) possono essere combinate in operazioni complesse o strutturate; la combinazione deve avvenire in forme determinate, chiamate strutture di controllo (la

---

<sup>7</sup>KNUTH D.E., *The Art of Computer Programming, I. Fundamental Algorithms*, cit., pp. 7-8; sul concetto di algoritmo, cfr. anche LANZARONE G.A., *Algoritmo*, in *Gli strumenti del sapere contemporaneo, Volume II, I concetti*, Grande dizionario enciclopedico, Utet, Torino, 1985, pp. 7-9. Sul concetto di algoritmo nell'informatica-giuridica, REISINGER L., *Approach to a Theory of "Soft Algorithms"*, in MARTINO A.A., SOCCI NATALI F., *Automated Analysis of Legal Texts*, North Holland, Amsterdam, 1986, pp. 761-772.

concatenazione, l'alternativa, la ripetizione)<sup>8</sup>. Esaminiamo le principali strutture di controllo<sup>9</sup>.

**Istruzione di assegnamento:** è l'istruzione fondamentale, specifica che un determinato valore calcolato va assegnato ad una variabile<sup>10</sup>. Ad esempio l'istruzione:

```
somma:=3+5
```

assegna alla variabile **somma** il valore 8 (cioè il risultato di 3+5).

**Istruzione composta:** specifica che le istruzioni che la compongono vanno eseguite l'una dopo l'altra, nell'ordine in cui sono scritte. Seguendo la convenzione del linguaggio PASCAL racchiudiamo le istruzioni componenti tra le espressioni **begin** (che segna l'inizio dell'istruzione composta) e l'espressione **end** (che ne segna la fine). Ad esempio, l'istruzione:

```
begin
    somma=3+5
    write(somma)
end
```

prescrive di assegnare il valore 8 alla variabile **somma** e di scrivere valore di **somma** (8) sul dispositivo di output dell'elaboratore (lo schermo).

**Istruzione condizionale (alternativa):** specifica che una certa istruzione va eseguita se e solo se è vera una certa espressione. Se l'espressione è falsa non viene eseguita nessuna istruzione o viene eseguita un'altra istruzione determinata. Ad esempio:

---

<sup>8</sup>Sulla programmazione strutturata cfr., tra gli altri, DIJKSTRA E.W., *A Discipline of Programming*, Prentice-Hall, Englewood Cliffs (NJ), 1976; WIRTH N., *Algorithms + Data Structures = Programs*, Prentice Hall, Englewood Cliffs (NJ); LANZARONE G.A., MAIOCCHI R., POLILLO R., *Introduzione alla programmazione strutturata. Il caso del Fortran, Cobol e Assembler*, Franco Angeli, Milano, 1981; CASADEI G., TEOLIS A., *Introduzione all'informatica. La programmazione*, Zanichelli, Bologna, 1979.

<sup>9</sup>In questi esempi ci si ispirerà al linguaggio di programmazione PASCAL, ma senza seguirne rigorosamente la sintassi.

<sup>10</sup>Nel contesto dei linguaggi di programmazione tradizionali, una variabile rappresenta sostanzialmente una zona di memoria nella quale possono essere immagazzinati determinati dati. Il dato memorizzato in una variabile viene chiamato il valore di quella variabile.

```

if reddito>6.600.000 then
    write ("devi presentare la dichiarazione dei
           redditi")
else
    write ("non devi presentare la dichiarazione dei
           redditi")

```

specifica che, qualora la variabile `reddito` abbia un valore superiore a 6.600.000, le parole "devi presentare la dichiarazione dei redditi" vengano scritte sullo schermo dell'elaboratore, se invece il valore di `reddito` è inferiore a 6.600.000 compare la scritta "non devi presentare la dichiarazione dei redditi".

**Istruzione iterativa (ripetizione):** specifica che certe istruzioni vanno eseguite ripetutamente, fino a che sia vera una certa espressione. Esempio

```

open (indirizzi)
while not eof(indirizzi)
begin
    read(indirizzo)
    write(indirizzo.cognome)
end

```

Questo algoritmo specifica che l'elaboratore, dopo aver aperto un archivio di indirizzi<sup>11</sup>, finché non è arrivato alla fine dell'archivio, ne legga una scheda (ne faccia una copia nella variabile `indirizzo`) e scriva sullo schermo il valore del campo `cognome` di quella scheda.

Breve esempio di programma strutturato

```

paga_oraria1:=20.000
paga_oraria2:=30.000
open(dipendenti)
while not eof(dipendenti)
begin
    if dipendente.qualifica=1 then
        paga=paga_oraria1*dipendente.ore_lavorate
    if dipendente.qualifica=2 then
        paga=paga_oraria2*dipendente.ore_lavorate
    write(paga)
end

```

Il programma specifica che dopo aver aperto l'archivio dipendenti, nel quale è registrato il nome, la qualifica e

---

<sup>11</sup>Un archivio o *file* è costituito da una sequenza di records o schede. I files risiedono di regola sulla memoria di massa (il disco). L'istruzione di apertura del file specifica quali files saranno usati dal programma.



il numero di ore lavorate da ogni dipendente, si legga ciascuna scheda, si calcoli lo stipendio (se il dipendente ha la qualifica 1, alla paga oraria di 1. 20.000, se ha la qualifica 2, alla paga oraria di 30.000), e si scriva il risultato sullo schermo.

La programmazione strutturata nasce dall'esigenza di controllare la complessità dei programmi informatici, di renderli più semplici, rigorosi, intersoggettivamente comunicabili. Infatti, le strutture sopra riportate sono sufficienti ad esprimere qualsiasi algoritmo, il loro uso consente di usare strumenti logici per controllare la correttezza dei programmi, un programma strutturato è più facilmente comprensibile e controllabile anche da parte di persone diverse da quelle che l'hanno scritto.

Diventano così possibili non solo nuovi metodi di programmazione ma anche nuove forme di divisione del lavoro.

Secondo il metodo top-down o del raffinamento progressivo, il problema che un sistema automatico deve risolvere (o, se si preferisce, il compito che deve svolgere) è scomposto nella combinazione - in una struttura di controllo - di un piccolo numero di sotto-problemi (ciascuno dei quali possa essere risolto più facilmente del problema iniziale). Questa scomposizione continua per ciascuno dei sottoproblemi finché non si giunge a problemi tanto semplici da poter essere risolti direttamente con un'istruzione del linguaggio di programmazione prescelto.

A questo metodo può corrispondere un'organizzazione del lavoro nella quale si distingue tra una attività di analisi (l'individuazione della struttura del programma e delle operazioni più astratte da svolgere: le cosiddette "specifiche") e l'attività di programmazione (la realizzazione dei programmi che svolgano le singole operazioni individuate nella fase di analisi, realizzazione che, per ciascuna di tali operazioni, può essere affidata ad un diverso programmatore). La programmazione strutturata facilita il processo di separazione dei diversi aspetti del lavoro informatico, consente di realizzare anche nell'informatica la separazione tra conoscenza e lavoro caratteristica dell'industrializzazione: gli aspetti creativi e intellettualmente stimolanti (la definizione dell'architettura del sistema, delle sue prestazioni e della sua struttura) si concentrano nella fase di analisi, mentre la fase di programmazione consiste nella "codifica" delle specifiche, cioè nella loro traduzione in un determinato linguaggio di programmazione, un'attività minuziosa che richiede molto tempo, ma spesso noiosa e meramente esecutiva.

Ulteriori progressi verso una migliore gestione di grandi quantità di dati si hanno con le basi o banche di dati. I sistemi per la gestione di basi di dati consentono all'utente - che può essere un uomo, o anche un programma informatico - "di considerare i dati in termini astratti anziché nel modo nel quale l'elaboratore li memorizza":

l'utente si limita a "specificare ciò che deve essere fatto, senza prestare attenzione al dettaglio degli algoritmi o della rappresentazione dei dati usati nel sistema"<sup>12</sup>.

L'uso di sistemi per la gestione di basi di dati semplifica enormemente l'uso dei dati registrati su un supporto magnetico e lo rende più indipendente dalle modifiche delle strutture nella quali i dati sono registrati nel supporto magnetico.

L'utente si limita a specificare le informazioni alle quali intende accedere, il sistema determina il modo o la procedura da seguire per rispondere all'interrogazione.

Infatti, il reperimento delle informazioni contenute in una base di dati richiede l'esecuzione di una serie di operazioni che sarebbe estremamente difficile per l'utente specificare nel dettaglio. E' il sistema di gestione della base di dati che "interpreta" i comandi dell'utente traducendoli in una sequenza di operazioni elementari sui dati, nel contesto dello specifico sistema di elaborazione<sup>13</sup>.

Si suole distinguere tra due tipi di programmi informatici per banche dati

-i sistemi per la gestione di basi di dati in senso stretto, che consentono la costituzione, la modificazione e

---

<sup>12</sup>ULLMAN J.D., *Principles of Databases Systems*, Computer Science Press, Potomac (Maryland), 1980, p. 1.

<sup>13</sup>Ad esempio, consideriamo il problema di reperire tutte le leggi in materia di tutela della riservatezza, compito che potremmo esprimere come segue nel linguaggio di interrogazione della banca dati Italgire (ci limitiamo a prendere in considerazione una ricerca in base alle parole contenute nel testo e supponiamo di trovarci nell'archivio delle leggi statali),

**\$PT: RISERVATEZZA;**

Per rispondere all'interrogazione il sistema deve

(a) ricercare la parola "riservatezza" negli indici relativi alle parole del testo, cioè gli elenchi che associano ad ogni parola, i codici che identificano i testi di legge che la contengono,

(b) reperire tutti i testi identificati dai codici associati a "riservatezza", e registrarli in una memoria temporanea.

Ognuna di queste azioni corrisponde a sua volta ad una sequenza di operazioni più semplici. Ad esempio, esaminare l'indice significa, finché non è stato esaminato tutto l'archivio indice, (mi limito ad illustrare una semplice ricerca sequenziale), per ogni registrazione (o record) dell'archivio,

(a) controllare se il campo che contiene la parola è composto di caratteri che coincidono con quelli del vocabolo "riservatezza"

(b) se il controllo sub (a) è positivo, memorizzare i numeri di identificazione.

Quanto più si approfondisce il dettaglio, tanto più le operazioni da compiere dipendono dalla struttura del sistema di elaborazione. Ad esempio, l'organizzazione dell'indice determina il modo nel quale questo può essere esaminato.

l'interrogazione dati strutturati (ad ogni oggetto o entità rappresentata nel sistema, sono associati determinati attributi con valori numerici o alfabetici determinati<sup>14</sup>;

-i sistemi di ricerca documentaria o di ricerca di informazioni (*information retrieval*), specificamente orientati alla gestione di dati testuali (cioè alla memorizzazione e l'interrogazione di testi integrali, riassunti, indicazioni bibliografiche ecc.).

### 3. I comportamenti intelligenti

L'informatizzazione delle due aree appena indicata si è conclusa con successo. Anche se non sempre sono stati raggiunti completamente i risultati sperati<sup>15</sup>, sembra non vi sia alcuna attività compresa in quelle aree che non sia stata affidata ad un sistema informatico, o non possa esserlo in futuro.

Si pensi, ad esempio, all'impatto dell'informatica sull'attività bancaria. La gestione dei dati relativi alle operazioni bancarie, l'aggiornamento delle situazioni creditizie, i trasferimenti di fondi vengono svolti oggi mediante l'elaboratore. Il lavoro si è modificato radicalmente: alcune attività che un tempo rientravano nelle mansioni dell'impiegato vengono svolte esclusivamente dal sistema informatico (così per i calcoli numerici relativi all'aggiornamento delle situazioni creditizie), altre - la maggioranza delle operazioni complesse - sono svolte nell'interazione tra impiegato e sistema informatico.

Recentemente l'informatica ha iniziato ad introdursi anche un nuovo settore di attività umana, quello dei comportamenti che richiedono intelligenza.

#### 3.1. L'intelligenza artificiale

La disciplina che sta elaborando gli strumenti per questa nuova fase della informatizzazione è la cosiddetta *intelligenza artificiale*, cioè la disciplina che, secondo la definizione comunemente accolta, si propone di realizzare strumenti informatici in grado di compiere attività che, se

---

<sup>14</sup>Ad esempio, in una base di dati relativa al personale di un'azienda, ogni dipendente avrà un nome, una data di nascita, una qualifica, una retribuzione oraria ecc.

<sup>15</sup>Ad esempio, si è molto parlato di *paperless offices*, nei quali dovrebbero essere eliminate le operazioni su supporto cartaceo, e gli scambi di informazioni dovrebbero svolgersi soprattutto tramite tecnologie informatiche, ma i tentativi di automazione più spinti delle attività d'ufficio, non hanno condotto agli aumenti di produttività che ci si attendevano.

fossero svolte da uomini, sarebbero attribuite alla loro intelligenza<sup>16</sup>.

L'ambito dell'intelligenza artificiale rischia, peraltro, di essere identificato mediante una categoria residuale. Infatti, per "comportamenti intelligenti", data la difficoltà e forse l'impossibilità di dare una definizione del concetto di intelligenza, si finisce per intendere tutto quello che un sistema automatico non è ancora in grado di fare<sup>17</sup>. I principali compiti di cui si è occupata l'intelligenza artificiale possono essere suddivisi in due grandi categorie<sup>18</sup>:

- compiti del senso comune<sup>19</sup> come la comprensione del linguaggio naturale (ad es. l'italiano), il riconoscimento di scene o di suoni, la comprensione di racconti, la manipolazione di oggetti, i giochi<sup>20</sup> ecc.

- compiti da esperti<sup>21</sup>, come la diagnosi medica, la progettazione di sistemi informatici, l'effettuazione di complessi calcoli matematici, la localizzazione di depositi minerari, il riconoscimento di strutture chimiche, il disegno ingegneristico<sup>22</sup>.

Il tentativo di realizzare un'estensione tendenzialmente illimitata del processo di informatizzazione:

---

<sup>16</sup>Marvin Minski, uno dei fondatori dell'intelligenza artificiale, la definisce come la disciplina che ha lo scopo di "far fare ai calcolatori attività che richiederebbero intelligenza se fatte dall'uomo"

<sup>17</sup>Cfr., per tutti RICH E., *Intelligenza artificiale*, McGraw Hill, Milano, 1986, p. 15, che definisce l'intelligenza artificiale come "lo studio che ha come scopo quello di mettere i calcolatori in grado di fare cose in cui, per il momento, gli esseri umani sono più abili".

<sup>18</sup>Cfr. RICH E., *Artificial Intelligence*, in SHAPIRO S.C., ECKROTH D., *Encyclopedia of Artificial Intelligence*, cit., p. 9.

<sup>19</sup>Che tutti gli uomini, in generale, sono in grado di affrontare adeguatamente, senza la necessità di una speciale formazione. L'intelligenza artificiale si occupa anche di funzioni, come la percezione visiva o uditiva, alle quali non siamo soliti associare l'idea dell'intelligenza, ma che è estremamente difficile far svolgere ad un elaboratore.

<sup>20</sup>Non sempre i giochi possono essere considerati compiti del senso comune. Giocare ad alto livello spesso richiede addestramento e attitudine, e quindi va collocato tra i compiti da esperti.

<sup>21</sup>Che possono essere svolti solo da persone dotate di formazione e competenza specifiche.

<sup>22</sup>Per un'ampia rassegna, cfr. BARR A., FEIGENBAUM E.A., COHEN P. R. (a cura di), *Handbook of Artificial Intelligence*, Kaufmann, Los Altos (California), 1981.

-da un lato sta promovendo lo sviluppo di nuovi strumenti informatici e, più in generale, un nuovo approccio all'informatica;

-dall'altro lato pone una serie di interrogativi di fondo, non solo tecnologici, ma anche filosofici ed etico-politici.

Lo sviluppo di applicazioni informatiche intelligenti richiederà forse nuovi strumenti hardware, un nuovo tipo di elaboratori elettronici<sup>23</sup>; ha già determinato una profonda revisione degli strumenti e dei metodi di programmazione, e più in generale, un nuovo rapporto tra studi umanistici e studi tecnologici.

Taluni autori hanno osservato che lo sviluppo di sistemi in grado di compiere attività intelligenti, l'informatizzazione di ciò che la nostra tradizione filosofica ritiene specificamente umano<sup>24</sup>, richiede una progressiva antropomorfizzazione dei sistemi informatici<sup>25</sup>. Sistemi capaci di svolgere attività che richiedono intelligenza possono essere sviluppati solo sulla base di uno studio approfondito dei metodi che l'uomo adotta per risolvere i problemi, nei diversi contesti naturali e sociali. Infatti, nell'intelligenza artificiale concorrono due nuove discipline:

-per gli aspetti umanistici, la c.d. scienza cognitiva, che rappresenta una combinazione di filosofia, linguistica e psicologia, viste in funzione della realizzazione di sistemi informatici;

-per gli aspetti ingegneristici, l'intelligenza artificiale in senso stretto, che si occupa di linguaggi e tecniche di programmazione.

La prospettiva di una progressiva estensione dell'informatizzazione, seppure sulla base dell'antropomorfizzazione degli elaboratori appena

---

<sup>23</sup> Sono stati realizzati elaboratori paralleli, in grado di eseguire contemporaneamente diverse istruzioni e elaboratori specificamente progettati per i linguaggi dell'intelligenza artificiale. Si tratta, in particolare, delle cosiddette *LISP machines*, in grado di eseguire con particolare efficienza programmi in LISP (uno dei linguaggio di programmazione più usati nell'intelligenza artificiale (cfr. par. 3.2), alle quali si affiancheranno forse speciali elaboratori per il linguaggio PROLOG (cfr. par. 3.2 e cap. 7, par. 2).

<sup>24</sup>Ricordiamo la definizione aristotelica dell'uomo come animale razionale, come essere la cui differenza specifica è il possesso della ragione. Cfr. ARISTOTELE, *La Politica*, Laterza, Bari, 1966, I, 2, 1253 a 9; VII, 13, 1332 b 5: "Ora gli altri animali vivono essenzialmente guidati da natura, taluni, ma entro limiti ristretti, anche dall'abitudine, e l'uomo pure dalla ragione perché egli solo possiede la ragione".

<sup>25</sup>Cfr., anche per alcuni riferimenti bibliografici, LANZARONE G.A., *Informatica*, cit., p. 397.

delineata, solleva una questione preliminare: questa prospettiva non nasce forse da un'immagine riduttiva dell'uomo, non si nasconde nello stesso lessico antropomorfo dell'informatica e, in particolare, dell'intelligenza artificiale<sup>26</sup> un errore di fondo, l'assimilazione di due realtà non comparabili tra loro, la nostra mente e il funzionamento di un sistema automatico?.

Il problema del rapporto tra le capacità degli uomini e quelle (attuali e future) degli elaboratori elettronici non è risolvibile mediante verifica empirica. Le diverse risposte fanno riferimento ad assunti e a problemi tanto fondamentali, e al tempo stesso indeterminati, ad approcci tanto incompatibili che è difficile trovare un metodo di controllo diverso dall'argomentazione filosofica. Tuttavia, è indubbio che talune ipotesi eccessivamente ottimistiche sulle prospettive dell'intelligenza artificiale si basano su un'immagine riduttiva dell'intelligenza. La lentezza e la difficoltà dei progressi confermano l'esigenza di una grande cautela nell'assimilare prestazioni degli uomini e degli elaboratori elettronici nei compiti che richiedono intelligenza. Possiamo ricordare, ad esempio che Newell e Simon, due pionieri dell'intelligenza artificiale (e scienziati di indubbio valore) affermavano nel 1957 che, entro dieci anni:

(a) Il campione del mondo di scacchi sarebbe stato un elaboratore;

(b) Un elaboratore avrebbe scoperto un nuovo teorema matematico importante;

(c) La maggior parte delle tesi nella psicologia avrebbero avuto la forma di programmi informatici, o di commenti a tali programmi.

A trent'anni di distanza nessuna delle predizioni appena menzionate si è realizzata<sup>27</sup>.

---

<sup>26</sup>Da tempo si parla di cervelli elettronici, che leggono da, o scrivono su una memoria (per dire che viene fatta una copia di un insieme di simboli su, o da, un supporto magnetico), che capiscono una certa domanda (quando la risposta del sistema è quella prevista), che interpretano un linguaggio ecc. Oggi si parla di intelligenza artificiale, di basi di conoscenza, di sistemi (esperti, di conoscenza) in grado di ragionare, di apprendere, di comprendere, di riconoscere ecc.

<sup>27</sup>Tra le profezie più inquietanti ricordiamo quella avanzata da GOOD I.J., *Speculations Concerning the First Ultraintelligent Machine*, in ALT F.L., RUBINOFF M., *Advances in Computer*, 6, Academic Press, New York, 1965, pp. 31-38 il quale riteneva probabile la costruzione entro il 2000 di una macchina "ultraintelligente", cioè tale da "superare tutte le attività intellettuali di ogni uomo, per quanto questi sia intelligente". Lo sviluppo questa macchina avrebbe costituito l'ultima invenzione dell'uomo, poiché anche nell'invenzione, e in particolare nell'invenzione di macchine intelligenti l'uomo sarebbe stato superato



La stessa ipotesi del sistema di simboli fisico, nella quale si suole ravvisare la premessa degli studi di intelligenza artificiale<sup>28</sup> può sembrare viziata da un certo riduzionismo. Si tratta dell'assunto che un sistema che disponga (a) di un insieme di schemi o simboli fisici e (b) di un insieme di meccanismi idonei a produrre una sequenza temporale di strutture costruite con questi simboli (l'elaboratore è lo strumento per costruire e manipolare queste strutture), sia capace dell'azione intelligente generalizzata<sup>29</sup>.

Infatti, taluni studiosi hanno messo in dubbio che l'operare della mente umana possa essere ricondotto ad operazioni di questo tipo, e hanno sostenuto che la considerazione degli aspetti contenutistici (dei significati), l'intenzionalità, la coscienza, il coinvolgimento dell'uomo nella realtà complessa dell'azione fisica e sociale, non sono riconducibili ad operazioni siffatte, e segnano una differenza insuperabile tra l'uomo e le macchine, o almeno tra l'uomo e il tipo di macchina cui si fa riferimento con l'espressione "elaboratore elettronico"<sup>30</sup>.

Altri hanno invece affermato che questa supposta differenza tra uomo e sistemi automatici non dipende dalla natura dei sistemi stessi, ma dal livello di astrazione ai quali essi vengono considerati. Ad un livello di astrazione basso tanto il cervello umano quanto l'elaboratore elettronico compiono operazioni elementari (sinapsi tra neuroni, operazioni del processore). Entrambi i sistemi (uomo e sistema informatico), visti nel loro complesso, ad un conveniente livello di astrazione, possono esibire un comportamento "intelligente"<sup>31</sup>. Si è anche osservato che la complessità del comportamento dell'uomo<sup>32</sup> sarebbe in gran parte dipendente dalla complessità dell'ambiente e non da

---

dalla macchina ultraintelligente, e dalle macchine sempre più intelligenti create da macchine di questo tipo.

<sup>28</sup>Cfr. NEWELL A., SIMON H.A., *Computer Science as Empirical Enquiry: Symbols and Search*, in "Communication of the ACM", Vol 19, n° 3, 1976 (marzo), pp. 279-293.

<sup>29</sup>Cfr. RICH E., *Intelligenza artificiale*, cit., p. 17 ss.

<sup>30</sup>Per una critica dell'intelligenza artificiale, cfr. SEARLE J.R., *Minds, Brains and Science*, The 1984 Reith lectures, British Broadcasting Corporation, Pitnam Press, Bath, 1984; DREYFUS H.L., DREYFUS S.E., *Mind over Machine*, Blackwell, Oxford, 1986; WINOGRAD T., FLORES C.F., *Understanding Computer and Cognition*, Ablex Publishing Co., Norwood (N.J.), 1986.

<sup>31</sup>Cfr., per tutti, HOFSTADTER D.R., DENNET D.C., *The Mind's I*, Basic Books, New York, 1981, p. 353 ss.

<sup>32</sup>In generale, il comportamento dei sistemi biologici.

una complessità interna. Anche il comportamento umano sarebbe riconducibile a pochi meccanismi relativamente semplici<sup>33</sup>.

Senza approfondire ulteriormente il problema (che riprenderemo più ampiamente esaminando le applicazioni informatico-giuridiche), possiamo osservare che, anche se taluni studiosi di intelligenza artificiale muovono da concezioni riduttive della vita e della ragione, ciò non deve far dimenticare che le ricerche e le applicazioni in questo settore hanno condotto ad risultati già significativi: alcuni aspetti della nostra intelligenza, sembrano simulabili in sistemi automatici. Si tratta delle attività che possiamo qualificare come applicative in senso ampio, cioè che usano uno schema concettuale dato senza metterlo in discussione (includiamo tra le attività applicative anche un certo grado di capacità di comprendere talune categorie di testi, di acquisire nuovi elementi di conoscenza, di risolvere problemi standardizzati ecc.); per le attività che richiedano invece intuizione e creatività capacità poco definibili, poco studiate e forse poco studiabili, sembra molto lontano, se mai arriverà, il momento nel quale gli uomini potranno essere sostituiti da sistemi informatici<sup>34</sup>.

### 3.2. I linguaggi dell'intelligenza artificiale

Come è noto, l'elaboratore è in grado di eseguire direttamente programmi formulati nel linguaggio macchina<sup>35</sup>, che risulta invece di difficile uso per l'uomo. A partire dal linguaggio macchina, nell'intento di facilitare la comunicazione tra uomo ed elaboratore sono stati elaborati linguaggi via via più perfezionati ed astratti: Assembler, Fortran, Basic, Algol, Pascal, C ecc. Tutti questi linguaggi presentano però un vizio fondamentale: sono stati sviluppati mediante un'astrazione progressiva a partire dal funzionamento degli attuali elaboratori elettronici, cioè a partire dal modello della macchina di von Neumann<sup>36</sup>. Da ciò deriva che, per essere rappresentato in questi linguaggi, un

---

<sup>33</sup>Cfr. SIMON H.A., *Le scienze dell'artificiale*, ISEDI, Milano, 1973 (traduzione di *The Sciences of Artificial*, The M.I.T. Press, Cambridge (Mass.), 1969, 40 ss., il quale enuncia la seguente tesi: "Un uomo, considerato come sistema capace di avere un comportamento, è semplicissimo. L'apparente complessità del suo comportamento nel tempo è in gran parte un riflesso della complessità dell'ambiente in cui egli si trova" (p. 43).

<sup>34</sup>Cfr. SOWA J.F., *Conceptual Structures*, cit., p. 350 ss.

<sup>35</sup>Formulate cioè nel linguaggio binario, come sequenze di 0 e 1.

<sup>36</sup>Cfr. introduzione al presente capitolo.

contenuto concettuale deve essere tradotto in una combinazione di operazioni elementari sui dati. Questa riduzione è difficile, ingiustificata, proprio per l'area dei problemi caratteristici dell'intelligenza artificiale.

I linguaggi dell'intelligenza artificiale possono essere ricondotti a due approcci fondamentali<sup>37</sup>:

-La programmazione funzionale. I linguaggi funzionali come il LISP<sup>38</sup>, si basano sul concetto logico-matematico di funzione<sup>39</sup>. La programmazione in LISP può essere interpretata fondamentalmente come composizione di funzioni. I programmi sono funzioni complesse, definite combinando funzioni semplici, e l'esecuzione del programma consiste nella determinazione del valore di queste funzioni.

-La programmazione logica. I linguaggi di programmazione come il PROLOG, si basano invece sulla logica simbolica. La conoscenza va rappresentata usando strutture logiche e va usata con metodi per la dimostrazione automatica di teoremi.

Entrambi questi approcci si caratterizzano quindi per il fatto di non derivare dalla macchina di von Neumann, ma da un modello astratto<sup>40</sup>.

Questi linguaggi consentono di dare un nuovo valore alla programmazione, soprattutto per le persone di formazione umanistica. La formalizzazione in un linguaggio di programmazione tradizionale cerca di esprimere dei contenuti concettuali nei termini del funzionamento di una macchina determinata, risponde a requisiti tecnologici piuttosto che all'esigenza della chiarificazione e dello sviluppo del pensiero.

La programmazione logica, ancor più della programmazione funzionale, risponde invece anzitutto (a prescindere dalla possibilità di un'implementazione informatica) all'esigenza di oggettivare e precisare dei contenuti concettuali in strutture linguistiche esattamente definite. Essa spesso contribuisce alla chiarificazione e allo sviluppo del pensiero.

---

<sup>37</sup>Un altro approccio di notevole interesse è rappresentato dalla c.d. programmazione ad oggetti.

<sup>38</sup>Il LISP fu inventato alla fine degli anni '50 da John McCarthy, presso il Massachusetts Institute of Technology.

<sup>39</sup>Come è noto, una funzione è una legge di corrispondenza univoca tra due insiemi **A** (dominio) e **B** (codominio) che associa ad ogni elemento di **A** (argomento) al massimo un elemento di **B** (valore)

<sup>40</sup>Cfr. STERLING L., SHAPIRO E., *The Art of Prolog*, MIT Press, Cambridge (Mass.), 1987, p. XVII.

### 3.3. Intelligenza artificiale, spazio di stati e sistemi assiomatici formali

Le incertezze, i problemi, la varietà degli studi, delle applicazioni e degli approcci che si connettono all'intelligenza artificiale non impediscono di individuare anche qui un paradigma fondamentale.

Un approccio generale per definire i tipici problemi di intelligenza artificiale è il modello della ricerca in uno spazio di stati. In questo modello, la definizione di un problema si compone di 3 parti:

- lo stato iniziale,
- lo stato o gli stati meta, che rappresentano l'obiettivo da raggiungere,
- le regole che definiscono le possibili transizioni da uno stato al successivo.

Ad esempio, il problema di vincere una partita di scacchi può essere definito come segue<sup>41</sup>:

**Stato iniziale:** la configurazione della scacchiera all'inizio della partita;

**Stati meta:** posizioni nelle quali il re avversario è in scacco matto:

**regole:** possibili mosse

Uno spazio di stati può essere rappresentato come un grafo. I nodi del grafo rappresentano stati del problema. Un arco che collega due nodi indica che c'è una regola di transizione (una mossa) che trasforma uno stato nell'altro. Il problema viene risolto con individuando un cammino (una sequenza di mosse) nello spazio di stati che collega lo stato iniziale ad uno degli stati meta.

Nel modello dello spazio di stati possiamo rappresentare anche dei ragionamenti (degli stati di conoscenza).

Ad esempio, possiamo rappresentare come segue il problema di determinare se un certo numero è maggiore di un altro<sup>42</sup>.

---

<sup>41</sup>RICH E., *Artificial Intelligence*, cit., p. 11.

<sup>42</sup>Nelle figure successive alla prima ci si limita a rappresentare le nuove conoscenze ottenute con la mossa che conduce alla figura di cui si tratta. La conoscenza rappresentata nella figura precedente, conoscenza che viene conservata nel nuovo stato, si indica con un asterisco.

**stato iniziale:**  
 (conoscenza che  
 ogni numero è  
 minore del suo  
 successore)

```
+-----+
| 1<2 |
| 2<3 |
| 3<4 |
| <.> |
| .   |
+-----+
```

**regola di trasformazione:**

**per ogni  $X, Y, Z$   $((X < Y) \ \& \ (Y < Z)) \Rightarrow (X < Z)$**

(se un numero  $X$  è minore di un numero  $Y$  e  $Y$  è minore di un numero  $Z$  allora  $X$  è minore di  $Z$ )

**stato meta:**

```
+-----+
|      |
| 3<6 |
|      |
| *   |
+-----+
```

**spazio di stati:**

```
+-----+
| 1<2 |
| 2<3 |
| 3<4 |
| <.> |
| .   |
+-----+
```

```
+-----+ +-----+ +-----+ +-----+ ... +-----+
| 1<3 | | 2<4 | | 3<5 | | 4<6 | | 8<10 |
| *   | | *   | | *   | | *   | | *   |
+-----+ +-----+ +-----+ +-----+
```

```
+-----+ +-----+ +-----+
| 1<4 | | 2<5 | | 3<6 |
| *   | | *   | | *   |
+-----+ +-----+ +-----+
```

Ogni nodo (rettangolo) è una possibile configurazione del problema (uno stato) e ogni linea che collega due nodi, rappresenta una possibile mossa. In ogni stato (che non sia uno stato finale), una o più mosse alternative consentono di

passare ad un nuovo stato. La soluzione del problema si ottiene trovando un cammino che vada dalla configurazione iniziale ad uno stato meta.

Il modello dello spazio di stati è estremamente generico. Tuttavia, quando si consideri l'esigenza che stati e mosse siano rigorosamente definiti (usando linguaggi formali come quelli menzionati nel paragrafo precedente), finisce per coincidere il paradigma più definito del sistema assiomatico.

Consideriamo infatti le seguenti caratteristiche dei sistemi di intelligenza artificiale:

(a) Si tratta di sistemi che "applicano", seppure in senso ampio, la conoscenza di cui dispongono.

(b) Anche nelle applicazioni più avanzate la conoscenza è considerata sotto il profilo sintattico, quale collezione di simboli formati in base a regole rigidamente definite e da sottoporre a trasformazioni determinate.

(c) Nonostante si sia spesso contrapposto il ragionamento comune, con la sua capacità di affrontare situazioni solo parzialmente definite, alla logica, tutti i sistemi sin qui realizzati (prescindendo forse da talune realizzazioni sperimentali largamente insoddisfacenti) si limitano a compiere operazioni di inferenza definibili in termini logico-matematici (seppure a partire da dati incerti), e anche per il futuro si può pensare più alla possibilità di simulare con strumenti logico-matematici taluni aspetti del ragionamento inesatto piuttosto che ad un'adozione di un diverso metodo di ragionamento in sistemi automatici<sup>43</sup>.

L'ambito in cui si "applicano" delle conoscenze usando regole di trasformazione definite sintatticamente è fondamentalmente quello della giustificazione logico-matematica, sicché si può forse affermare che il modello al quale fanno riferimento le applicazioni di intelligenza artificiale (almeno sino ad oggi) quello logico-matematico del sistema deduttivo. In questa prospettiva i possibili stati sono quelli deducibili nel sistema assiomatico che risulta dalla rappresentazione formale della situazione iniziale e delle regole di trasformazione. L'applicazione di una regola di trasformazione che genera un nuovo stato può essere interpretata come un'inferenza logica.

---

<sup>43</sup>Sul rapporto tra logica e senso comune è da tempo in corso un vivace dibattito. Alcuni autori, (come MINSKI M., *A Framework for the Representation of Knowledge*, in WINSTON P.H., *The Psychology of Computer Vision*, McGraw Hill, New York, 1975, pp. 211-280 (già pubblicata come *Artificial Intelligence Memo 306*, MIT AI Lab, 1974)), hanno affermato l'inadeguatezza della logica rispetto al ragionamento comune, e hanno cercato di individuare metodi diversi per la rappresentazione e la manipolazione della conoscenza. Ritourneremo più ampiamente su questo problema nel seguito (cfr. cap. 9).

Come vedremo<sup>44</sup>, i sistemi informatici intelligenti consistono fondamentalmente di una base di conoscenza e di un motore inferenziale. La base di conoscenza può essere considerata la base assiomatica del sistema (l'insieme degli enunciati primitivi), mentre il motore inferenziale è un programma che deduce dalla base assiomatica ulteriori enunciati del sistema (i teoremi).

<b>Sistema deduttivo</b>	<b>Sistema di conoscenza</b>
-----	-----
<b>assiomi</b>	<b>base di conoscenza</b>
<b>regole di inferenza</b>	<b>motore inferenziale</b>
<b>enunciati deducibili nel sistema (conclusioni)</b>	<b>risposte (soluzioni di problemi)</b>

Come ha rilevato Kowalski<sup>45</sup>, in un algoritmo si possono distinguere due componenti fondamentali, la logica e il controllo, cioè la descrizione del problema da risolvere e la specificazione del metodo per risolvere quel problema:

-Nella programmazione tradizionale (chiamata programmazione algoritmica, procedurale o imperativa), anche se strutturata, le due componenti sono inscindibili, si fondono nella descrizione/prescrizione di una sequenza di operazioni sui dati. Chi intenda sviluppare un sistema informatico deve assumersi il compito gravoso di tradurre il problema originario nei termini di una combinazione di queste operazioni.

-Nell'intelligenza artificiale, invece, le due componenti sono separate: la conoscenza necessaria per risolvere il problema va rappresentata formalmente in una base di conoscenza; le procedure per manipolare la conoscenza vanno implementate nel motore inferenziale. E' possibile sviluppare motori inferenziali di impiego generale (in particolare, che usino le regole di inferenza della

<sup>44</sup>Cfr. cap. 4.

<sup>45</sup>Cfr. KOWALSKI R.A., *Logic for Problem Solving*, North Holland, New York, Oxford, 1979, p. 125 ss.

logica). Pertanto, per sviluppare un'applicazione informatica, l'uomo può limitarsi a rappresentare il problema e il suo contesto, a fornire la conoscenza c.d. dichiarativa; spetta invece al sistema (al motore inferenziale) determinare la procedura mediante la quale il problema potrà essere risolto<sup>46</sup>.

### 3.4. I limiti del paradigma del sistema assiomatico formale

Con riferimento al paradigma del sistema deduttivo possiamo evidenziare i limiti dell'intelligenza artificiale.

Il metodo deduttivo, è stato usato sin qui nel contesto della giustificazione, nel contesto cioè nel quale le scoperte (i teoremi), vengono giustificate, rese intersoggettivamente convincenti, riconducendole ad altri enunciati accolti come assiomi o già dimostrati come teoremi. Tipicamente una dimostrazione in un determinato sistema logico consiste in una sequenza finita di passi, ciascuno dei quali costituisce applicazione di una delle regole di inferenza del sistema, sequenza che ci consente di passare da un insieme di enunciati assunti come premesse ad un ulteriore enunciato, la conclusione.

I sistemi informatici intelligenti nascono forse da un uso "improprio" del modello del sistema assiomatico formale, dal suo trasferimento dal contesto della giustificazione al contesto della scoperta (intendendosi per scoperta la individuazione di conclusioni nuove). Ci si è infatti accorti che i meccanismi deduttivi potevano essere usati non solo per controllare se un certo enunciato sia deducibile o meno dalle premesse, ma anche per trovare gli individui per i quali, date certe premesse, risulti vero un determinato predicato, tra i quali valgano certe relazioni.

Esaminiamo, ad esempio, il funzionamento del sistema PROLOG, che possiamo considerare un semplice modello di sistema deduttivo automatico<sup>47</sup>.

Un programma PROLOG consiste fondamentalmente nella rappresentazione formale della conoscenza rilevante per il problema da risolvere. Tale conoscenza va tradotta in un insieme di assiomi espressi in un sottoinsieme della logica

---

<sup>46</sup>Nel seguito, si parlerà di programmi, metodi, approcci algoritmici, contrapponendoli all'approccio dichiarativo caratteristico dell'intelligenza artificiale. E' opportuno precisare che si tratta solamente di linee di tendenza. In particolare, anche chi sviluppi applicazioni di intelligenza artificiale, non potrà prescindere del tutto dalla considerazione dell'aspetto procedurale.

<sup>47</sup>Sul PROLOG, cfr. cap. 7 par. 2.



dei predicati, le clausole di Horn. Ad esempio, potremo definire come segue alcuni rapporti di parentela<sup>48</sup>:

P è padre di F se  
 P è genitore di F e  
 P è maschio.

M è madre di F se  
 M è genitore di F e  
 M è femmina.

N è nonno di Nip se  
 N è padre di F e  
 F è genitore di Nip.

N è nonna di Nip se  
 N è madre di F e  
 F è genitore di Nip.

Possiamo rappresentare come segue alcuni fatti relativi ai rapporti familiari tra divinità greche:

Zeus è genitore di Ares.  
Era è genitore di Ares  
Afrodite è genitore di Armonia.  
Zeus è maschio.  
Era è femmina.  
Afrodite è femmina.  
Ares è maschio.

Un volta che si siano rappresentate in questo modo le conoscenze in materia di rapporti di parentela tra divinità greche, possiamo porre domande al sistema.

Il sistema procederà cercando di dimostrare l'enunciato sul quale verte la domanda, che esprime il problema da risolvere. Esso cercherà di costruire una dimostrazione che partendo dalla base di conoscenza termini nell'enunciato di cui trattasi.

Se questo enunciato è specifico, (ad esempio Zeus è nonno di Armonia) il sistema risponderà con un "sì" o un no (a seconda che riesca o no a dimostrare l'enunciato). Se l'enunciato è generico (ad esempio, esiste un X che sia nonno di Armonia?) la dimostrazione sarà costruttiva: il sistema identificherà (se possibile) un individuo che

<sup>48</sup>Seguo la seguente convenzioni: le espressioni composte da una sola lettera alfabetica e inoltre tutte le sequenze di caratteri che iniziano con il trattino basso "\_" rappresentano variabili; le espressioni sottolineate rappresentano dei predicati (per i concetti di variabile e predicato, cfr. cap. 7, par. 1.2).

presenti le proprietà specificate nella domanda (il nonno di Armonia). In nome di ciascuno di tali individui (Zeus o Era) sarà una risposta alla domanda.

Il metodo seguito per individuare una dimostrazione consiste nell'applicazione metodica delle regole di inferenza agli assiomi (alla conoscenza formalizzata nel programma), in base ad una strategia predefinita.

Pur nell'ambito del ragionamento deduttivo, bisogna distinguere tre momenti:

- l'individuazione di un enunciato interessante, che sembri deducibile dalle premesse;
- la costruzione di una dimostrazione di quell'enunciato;
- il controllo della correttezza della dimostrazione.

Per un uomo il metodo dell'applicazione sistematica delle regole di inferenza

-può essere applicabile al controllo della correttezza di una dimostrazione data;

-è applicabile solo in limiti estremamente ristretti al problema della costruzione di una dimostrazione<sup>49</sup>.

Un elaboratore elettronico può svolgere talune manipolazioni simboliche con altissime velocità e precisione, può usare il metodo dell'applicazione sistematica di regole di inferenza esattamente definite ben al di là delle capacità umane. In questo modo, pur senza usare gli stessi metodi di ragionamento, un sistema automatico può raggiungere, in taluni settori, prestazioni analoghe a quelle dell'uomo. Così, se il campione del mondo di scacchi non è un elaboratore, vi sono tuttavia elaboratori che giocano a scacchi ad alto livello, pur usando "ragionamenti" completamente diversi da quelli degli scacchisti umani.

Tuttavia, l'ambito dei problemi che possono trovare una risposta con questi metodi trova dei precisi confini. Si è già osservato che il contesto della scoperta è scarsamente studiato e forse scarsamente studiabile, in esso entrano in considerazione l'intuizione e la creatività, elementi irriducibili ad ogni definizione. L'intelligenza artificiale sembra aver scelto, almeno per ora, una strada diversa e, pertanto, il processo di antropomorfizzazione

---

<sup>49</sup>Come è noto, questo metodo è applicabile alla logica proposizionale (con le tavole di verità o altre tecniche analoghe). Nel campo della logica dei predicati del primo ordine, e ancor più in quello delle teorie matematiche e scientifiche, che usano logiche di ordine superiore, anche dove un'applicazione automatica di regole di inferenza potrebbe dare dei risultati (a prescindere dai problemi dell'indecidibilità della logica dei predicati e dell'incompletezza dell'aritmetica elementare, sui quali non è questo il luogo per soffermarci), questa tecnica è troppo pesante, richiede troppe operazioni di calcolo, per essere applicata da un uomo in tempi ragionevoli.

dell'informatica<sup>50</sup> è limitato. Al paradigma del sistema deduttivo formale, la particolare tecnica adottata sin qui dall'informatica per simulare comportamenti intelligenti, si riconnettono problemi determinati, che considereremo nei paragrafi seguenti.

### 3.4.1 Il problema dell'esplosione combinatoria

Un primo problema è quello della cosiddetta esplosione combinatoria. L'applicazione sistematica delle regole di inferenza di un sistema deduttivo conduce all'individuazione di un numero così grande di alternative da esaminare per trovare la soluzione del problema, da costituire un limite concettuale (indipendente non solo dalle attuali ma anche dalle future possibilità tecnologiche) alla possibilità di un loro esame esaustivo. Agli assiomi possono essere applicate più regole di inferenza, ciascuna delle quali conduce ad un nuovo enunciato; ciascun nuovo enunciato può consentire l'applicazione di più regole di inferenza, così da giungere ad altri nuovi enunciati, e così via. Anche rispetto ad un problema ben delimitato come quello del gioco degli scacchi, vi è un numero enorme di stati diversi (di configurazioni della scacchiera risultanti dalle possibili mosse) da prendere in considerazione per scegliere la mossa migliore (circa 10 elevato alla 120).

Al problema della esplosione combinatoria si è tentato di porre rimedio mediante le euristiche, cioè mediante tecniche volte a migliorare l'efficienza del processo di ricerca, spesso sacrificandone la completezza. Non si esaminano indiscriminatamente tutti i cammini possibili, ma si esaminano solo (o prima) quelli che si sembrano più promettenti sulla base di un criterio, o funzione euristica, determinato.

Le tecniche euristiche seppure talvolta indispensabili (le rende necessarie l'esplosione combinatoria) e spesso opportune (in molte occasioni non è necessario trovare la soluzione ottima, basta un metodo che conduca ad una soluzione soddisfacente), possono sacrificare quelle caratteristiche di completezza e sistematicità della ricerca che rappresentano i principali vantaggi dell'uso di un sistema automatico: le euristiche comportano la possibilità dell'errore. Ad esempio, l'adozione di una tecnica euristica potrebbe condurre il sistema che gioca a scacchi a tralasciare i cammini che conducono alla vittoria, e quindi a perdere la partita.

Chi usa un sistema informatico in contesti così complessi da richiedere tecniche euristiche deve accettare la possibilità che il sistema si sbaglia, e deve essere consapevole del fatto che il sistema non possiede l'elasticità e la creatività che consentono spesso all'uomo di evitare l'errore (ad esempio, inventando una nuova

---

<sup>50</sup>Sul quale ci siamo soffermati nel par. 3.1.

tecnica più adatta alle specifiche circostanze, o riprendendola da contesti analoghi, anche se apparentemente lontani e privi di apparenti relazioni con quello nel quale si è presentato il problema), o comunque di reagire all'errore nel modo più opportuno, e di imparare dall'insuccesso<sup>51</sup>.

La possibilità dell'errore da parte di sistemi automatico "intelligente", e la limitatezza dei mezzi di cui questi dispongono per farvi fronte, impone che l'uso di tali sistemi sia circondato da particolari cautele, specialmente negli ambiti (ad esempio, la medicina) dove l'errore può avere conseguenze irreparabili. Non bisogna dimenticare però, che anche l'uomo sbaglia frequentemente. Spesso l'errore è determinato dalla mancanza di una diligente applicazione, in base a metodi prestabiliti, delle conoscenze di cui si dispone (è noto quanti errori derivino da negligenze o disattenzioni; inoltre, spesso mancano il tempo o i mezzi, per una valutazione tempestiva di un contesto complesso). L'uso di sistemi informatici delinea pertanto, accanto a gravi rischi, la possibilità di un'utile collaborazione tra uomo e sistemi automatici, anche in problemi difficili, per i quali non si disponga di un metodo in grado di fornire sempre una soluzione corretta.

### 3.4.2 Il problema della formalizzazione

Un secondo problema è quello della necessità di una formalizzazione. La formalizzazione, l'uso di un linguaggio precisamente definito<sup>52</sup>, è necessaria in quanto le regole di inferenza di un sistema formale (in altri termini, le regole di trasformazione da applicare ad un sistema simbolico) sono sintatticamente definite: esse trovano applicazione in quanto sussistano premesse (stati) dotati di determinati requisiti sintatticamente identificabili<sup>2</sup>.

Questo problema, sebbene possa sembrare non particolarmente grave in campi in una certa misura già assiomatizzati e formalizzati, è invece fondamentale nei contesti caratterizzati da esperienze complesse, difficilmente suscettibili di una rappresentazione univoca, esaustiva e definitiva (come ad esempio nelle scienze

---

<sup>51</sup>Allo stato della tecnica, i sistemi informatici sono in grado di apprendere dall'esperienza solo in una misura molto limitata.

<sup>52</sup>Ad esempio, in base alla regola di inferenza dell'eliminazione della implicazione o del *modus ponens* dalle premesse:

1.  $A \rightarrow B$  e
  2.  $A$ ,
- è deducibile l'enunciato  $B$ .

Ma non potremmo effettuare l'inferenza se il simbolo che unisce i due enunciati non fosse  $\rightarrow$  o se la condizione dell'implicazione e la seconda premessa non coincidessero sintatticamente (non fossero la stessa sequenza di caratteri).

mediche e ancor più nelle scienze sociali e, in generale, nel mondo della prassi, nelle discipline politiche, economiche, giuridiche). Nel mondo della politica, degli affari, del diritto, è necessario prendere decisioni anche in contesti scarsamente strutturati e in continua evoluzione, sulla base di informazioni incomplete, tenendo conto di un insieme di obiettivi non gerarchizzabili, talvolta contrastanti, spesso impliciti nelle nostre organizzazioni sociali. Ogni rappresentazione formale di un contesto di questo tipo, che sia sufficientemente precisa da consentire di raggiungere conclusioni determinate applicando metodi di inferenza automatica, non può pretendere di esaurire la realtà, di rappresentare con completezza e fedeltà il problema, non può che essere provvisoria e suscettibile di continue revisioni.

Come si vedrà nel seguito, si stanno sviluppando sistemi informatici in grado di trattare in una certa misura il linguaggio naturale, sistemi che operano una traduzione automatica di un testo in linguaggio naturale in strutture sintattiche precisamente definite, ma questa traduzione, benché possa risultare utile a fini determinati, non sostituisce il linguaggio naturale. Infatti questa traduzione nella misura in cui realizzi un sistema assiomatico che funga da base di conoscenza - postula proprio l'eliminazione di quelle caratteristiche del linguaggio naturale (non solo la vaghezza e l'ambiguità, ma soprattutto la sua complessità, il suo rapporto con l'esperienza e la cultura) che ne fanno lo strumento più appropriato per affrontare contesti come quelli appena menzionati.

Tuttavia, anche l'uomo, per decidere la propria azione in contesti siffatti, è costretto ad operare delle scelte: a conferire una interpretazione determinata, ad eventi e circostanze, completando, sulla base di ipotesi verosimili le informazioni insufficienti; ad attribuire la prevalenza ad alcuni tra i valori rilevanti stabilendo una gerarchia determinata tra di essi. Nella misura in cui le premesse di fatto e quelle normative si appoggino su una corretta argomentazione, esse sono tali da potere (da dovere) essere riconosciute anche in circostanze analoghe. Tali decisioni conducono (o almeno dovrebbero condurre) alla costruzione di una sorta di sistema assiomatico dal quale segua logicamente la soluzione prescelta, sistema assiomatico nel quale sono "oggettivate" le scelte che abbiamo appena menzionate<sup>53</sup>. Come rispetto all'esplosione combinatoria, anche rispetto alla formalizzazione si delineano sia (a) un grave pericolo, che (b) un'interessante prospettiva:

(a) Le scelte che hanno condotto ad una formalizzazione determinata potrebbero continuare a determinare il funzionamento del sistema automatico anche al di fuori del

---

<sup>53</sup>Secondo un'autorevole opinione, ogni decisione razionale deve poter essere giustificata in un contesto deduttivo (cfr. cap. 6, par. 3.2).

contesto nel quale quelle scelte trovano un fondamento, anche quando le ragioni che le hanno giustificate si rivelino errate, siano venute meno, o dovrebbero comunque essere riconsiderate. Privo di sufficienti capacità di critica e revisione, il sistema continuerebbe ad operare ciecamente sulla base del modello di cui dispone, con conseguenze imprevedibili e inaccettabili.

(b) La possibilità di rappresentare, di oggettivare determinate scelte in un sistema che le applichi automaticamente, può rivelarsi invece un fattore di razionalizzazione dell'attività decisionale. Nel predisporre un sistema informatico che serva da ausilio alla decisione, è infatti necessario acquisire consapevolezza dei processi decisionali, delle loro premesse e dei loro metodi; dovendo il sistema essere applicato ad una pluralità di casi è necessario uno sforzo nella direzione di una maggiore universalità. La realizzazione di un sistema automatico può migliorare, in particolare, le decisioni di routine, singolarmente di scarsa importanza, per le quali non c'è il tempo o l'interesse per un esame accurato, ma che, considerate nel loro insieme, possono rivestire una notevole importanza sociale.

## CAP. 2

### L' AUTOMAZIONE DEL LAVORO GIURIDICO: DALLA GIURIMETRIA ALL' INTELLIGENZA GIURIDICA ARTIFICIALE

Anche nell' informatizzazione del mondo del diritto è possibile ricostruire il processo delineato nelle pagine precedenti, la progressiva occupazione da parte dell' informatica di settori prima riservati al lavoro dell' uomo. Distinguerò due diversi concetti di lavoro giuridico:

-lavoro giuridico in senso ampio, che comprende tutta l'attività consistente in, o necessaria per, l'interpretazione e l'applicazione del diritto<sup>1</sup>;

-lavoro giuridico in senso stretto, che comprende la parte di quell'attività che presuppone, di regola, una specifica formazione giuridica di livello universitario.

Inoltre, userò la locuzione informatica giuridica per indicare, in generale, lo studio dell'uso dell'informatica nel mondo del diritto.

Attività informatizzata	Strumenti informatico-giuridici
calcoli quantitativi	giurimetria
gestione di dati	Automazione di procedure giuridiche, buroatica, sistemi di documentazione giuridica
attività intelligenti	Sistemi basati sulla conoscenza per l'analisi, la pianificazione giuridica, la ricerca di informazioni concettuale.

---

<sup>1</sup>Su interpretazione e applicazione del diritto, cfr. TARELLO G., *L'interpretazione della legge*, cit., p. 42 ss. Nel concetto di applicazione del diritto non ricomprendo l'uso di conoscenze tecniche (conoscenze ingegneristiche, chimiche, fisiche, psicologiche ecc.) che, pur necessarie per determinare una qualificazione giuridica, rientrano nella competenza di figure professionali diverse dal giurista.

## 1. Operazioni di calcolo

La disciplina dalla quale prese avvio l'informatica giuridica fu la giurimetria (*Jurimetrics*) con la quale ci si proponeva, di applicare metodi quantitativi nel mondo del diritto<sup>2</sup>.

Con il termine "giurimetria", in relazione agli sviluppi delle applicazioni informatico-giuridiche, vennero intese ben presto una serie di ricerche non completamente riconducibili all'etimo di quel termine: l'applicazione di modelli logici a norme giuridiche, l'uso dell'elaboratore all'attività giuridica, la previsione delle sentenze<sup>3</sup>. Già in lavori pubblicati negli anni 50<sup>4</sup> e negli anni 60<sup>5</sup> possiamo trovare l'anticipazione di molte delle idee alla base di sistemi informatico-giuridici che solo oggi vengono realizzati o proposti.

Infatti, la complessità delle problematiche giuridiche ha sempre spinto i migliori studiosi a guardare al di là dei modelli informatici dominanti, anche se la realizzazione di applicazioni funzionanti ha dovuto attendere che fossero predisposte tecnologie adeguate.

In particolare, un'informatica giuridica che consista nell'eseguire operazioni di calcolo a partire da dati giuridici, non può che essere di limitata applicazione. Vi sono operazioni di calcolo che costituiscono applicazione di norme giuridiche (ad esempio, il calcolo delle imposte) o che hanno ad oggetto dati giuridici (ad esempio, la statistica giudiziaria), o comunque rilevanti per il diritto, ma si tratta in genere, di operazioni relativamente semplici, che non presentano uno speciale interesse scientifico e che debbono essere automatizzate soprattutto in quanto debbano effettuarsi in relazione ad un grande

---

<sup>2</sup>Cfr. LOEVINGER L., *Jurimetrics: The Next Step Forward*, in "Minnesota Law Review", XXXIII, 1949, p. 455 ss.. Sulla giurimetria, cfr. LOSANO M., *Informatica per le scienze sociali*, cit., p. 41 ss.

<sup>3</sup>Cfr. BAADE H. W. (a cura di), *Jurimetrics*, Basic Books, New York, 1963 (originariamente pubblicato in "Law Contemp. Probl.", 28, pp. 1-270, 1963)

<sup>4</sup>Cfr. KELSO L.O., *Does the Law Need a Technological Revolution*, in "Rocky Mountains Law Review", 18, 1946, pp. 376-392; LOEVINGER L., *Jurimetrics. The Next Step Forward*, cit.

<sup>5</sup>Cfr. MEHL L., *Automation in the Legal World. From the Machine Processing of Legal Information to the 'Law Machine'*, in *Mechanisation of Thought Processes. National Physical Laboratory Symposium No 10*, novembre 1958., Her Majesty's Stationery Office, London, 1959, pp. 757-787.



numero di casi dello stesso tipo. L'esigenza di una loro automazione (al di là dei semplici calcoli che possono effettuarsi con una calcolatrice automatica) si pone nell'ambito di una più generale gestione di dati.

## 2. La gestione di dati

L'informatizzazione del diritto almeno nelle realizzazioni operative è ancor oggi limitata alla gestione di dati, anche se l'interesse teorico si è già spostato verso l'intelligenza artificiale. Nel settore della gestione di dati giuridicamente rilevanti possiamo ricomprendere fondamentalmente tre tipi di applicazioni:

- (a) l'applicazione automatica di norme giuridiche.
- (b) l'automazione del lavoro d'ufficio;
- (c) la gestione di basi di dati giuridici.

Si tratta di attività che toccano solo marginalmente l'attività del giurista (il lavoro giuridico in senso stretto), ma che possono essere fatte rientrare nella nozione più ampia di lavoro giuridico sopra introdotta.

### 2.1. L'applicazione automatica di norme giuridiche

Vi sono numerosi sistemi informatici tradizionali, anche di grosse dimensioni, che compiono elaborazioni che consistono nell'applicazione di norme giuridiche (comprendendo nella disciplina giuridica anche quella di origine negoziale). Nelle pagine seguenti si illustreranno due categorie di sistemi di questo tipo (1) sistemi nei quali il trattamento di aspetti giuridici è ricompreso in applicazioni gestionali, e (2) sistemi che intendono fornire una sorta di "consulenza giuridica" automatica. Si considereranno, infine, le prospettive e i limiti dell'applicazione di norme giuridiche mediante strumenti informatici tradizionali.

#### 2.1.1. Applicazioni gestionali

Alcune importanti applicazioni informatiche di tipo gestionale comprendono l'effettuazione di operazioni regolate dal diritto. Si pensi ai sistemi che hanno per oggetto:

- la determinazione e il pagamento delle retribuzioni, delle pensioni, dei contributi previdenziali;
- la determinazione e la riscossione di determinate categorie di tributi;
- la concessione periodica di sussidi;

-la gestione dei conti bancari o di varie forme di contabilità;

-la tenuta di vari registri (anagrafici, immobiliari, dei beni mobili registrati ecc.).

Di solito, tali sistemi affiancano alla gestione di archivi di dati (che nei sistemi più evoluti sono integrati in una base di dati) l'effettuazione di una serie di elaborazioni che usano quegli archivi: calcoli numerici, redazione di documenti di vario tipo (ad esempio, distinte, ricevute, mandati di pagamento, certificati) ecc. I programmi per queste elaborazioni di regola sono scritti in linguaggi di tipo tradizionale, e in particolare in COBOL, il linguaggio più usato per applicazioni gestionali.

Si tratta di sistemi spesso di grandi dimensioni, che trattano enormi quantità di dati, con elaborazioni talvolta prevalentemente in modalità a lotti<sup>6</sup> (tipicamente, la gestione degli stipendi, che è effettuata mensilmente), talvolta prevalentemente in modalità interattiva (come nella gestione dei conti bancari).

Sistemi informatici siffatti costituiscono un'elemento insostituibile in tutti i grandi apparati organizzativi, pubblici e privati. Per molte delle applicazioni summenzionate, il passaggio a tecnologie informatiche si configura ormai come una scelta irreversibile.

### 2.1.2. Sistemi per consulenze giuridiche

I sistemi che presenteremo in questo paragrafo si propongono di fornire una sorta di consulenza giuridica automatica e, quindi, prestazioni analoghe a quelle delle applicazioni giuridiche dell'intelligenza artificiale<sup>7</sup>.

Si tratta di programmi realizzati con tecniche informatiche tradizionali. Per quanto riguarda i linguaggi di programmazione: di solito anziché il COBOL si usano linguaggi più adatti allo sviluppo di piccole applicazioni, come BASIC, C o PASCAL. Infatti, per lo più, questi programmi sono destinati a microelaboratori.

I sistemi che stiamo considerando sono, in genere, interattivi: rivolgono una serie di domande predeterminate all'utente e la risposta a ciascuna di tali domande

---

<sup>6</sup>Un elaboratore opera in modalità a lotti (o come anche si dice *batch*) quando i lavori (un lavoro o *job* è il programma per svolgere una certa elaborazione, assieme ai dati relativi) vengono eseguiti solo dopo essere stati introdotti completamente nel sistema di calcolo. L'elaboratore memorizza in una lista o coda i diversi lavori che gli sono stati proposti e li esegue uno dopo l'altro, senza che l'utente possa intervenire.

<sup>7</sup>Cfr. cap. 4.

determina la successiva esecuzione del programma e quindi la prosecuzione dell'interazione con l'utente.

Tra i sistemi di tipo algoritmico progettati per dare consulenze giuridiche possiamo menzionare, a titolo di esempio, i seguenti:

-L'Automa infortunistico, sviluppato presso l'Istituto per la documentazione giuridica di Firenze verso la metà degli anni 70. Il programma, scritto in BASIC, aveva per oggetto la determinazione dell'indennizzo nel caso di incidenti stradali. Nel programma erano stati incorporati criteri di origine legislativa e giurisprudenziale. L'utente era chiamato a fornire una serie di informazioni, in base alle quali il sistema procedeva al calcolo dell'indennizzo<sup>8</sup>.

-I programmi CORPTAX<sup>9</sup>, CHOOSE<sup>10</sup> e SEARCH<sup>11</sup>, sviluppati da R. Hellawell all'inizio degli anni 80. Si tratta di sistemi realizzati in BASIC che trattano la normativa fiscale. Il sistema pone determinate domande all'utente, questi, in risposta, fornisce dei dati, in base ai quali il sistema provvede alla determinazione dell'importo dovuto.

-I programmi in BASIC per la gestione della normativa fiscale sviluppati da Bellord<sup>12</sup>.

-Il progetto DataLex<sup>13</sup>, che si propone di sviluppare un guscio (LES-Legal Expert System) per applicazioni informatico-giuridiche. L'utente deve predisporre ciò che gli autori chiamano reti di decisione (decision nets o decision networks) che, in sostanza, rappresentano dei

---

<sup>8</sup>Cfr FAMELI E., *L'automa infortunistico: un esperimento di consulenza giuridica automatica*, in "Informatica e diritto", 1, 1976, pp. 1-50.

<sup>9</sup>HELLAWELL R., *A Computer Program for Legal Planning and Analysis: Taxation of Stock Redemptions*, in "Columbia Journal of Transnational Law", 19, 1980, pp. 1363-1398.

<sup>10</sup>HELLAWELL R., *CHOOSE: A computer program for legal planning and analysis*, in "Akron Law Review", 15, 1981, pp. 339-357.

<sup>11</sup>HELLAWELL R., *SEARCH: A Computer Program for Legal Problem Solving*, in "Akron Law Review", 15, 1982, pp. 635-653.

<sup>12</sup>BELLORD N.J., *Tax Planning by Computer*, in NIBLETT B. (a cura di), *Computer Science and Law: An Advanced Course*, Cambridge University Press, 1979, pp. 173-182.

<sup>13</sup>GREENLEAF G., MOMBRAY A., TYREE A.L., *Expert systems in law: The DATALEX Project*, in *Proceedings of the First International Conference on Artificial Intelligence and Law*, cit., pp. 9-17.

diagrammi di flusso<sup>14</sup>. Sulla base delle reti di decisione vengono redatti automaticamente programmi nel linguaggio C.

### 2.1.3. Programmazione tradizionale e applicazione del diritto.

In tutte le applicazioni che abbiamo presentato nei paragrafi precedenti, attività giuridicamente rilevanti sono formalizzate nei termini di una combinazione di operazioni elementari sui dati, mediante un linguaggio di programmazione tradizionale (un linguaggio algoritmico). La conoscenza giuridica non è distinta dal programma che la usa, ma è implicita, incorporata nella prescrizione/descrizione delle operazioni che l'elaboratore deve compiere per raggiungere il risultato voluto. Il programma rappresenta (in termini di operazioni sui dati) un particolare procedimento (o procedura) per adempiere alla legge o per giungere alla determinazione di una qualificazione giuridica.

L'esame delle applicazioni che abbiamo appena illustrato, sia di quelle "gestionali", sia di quelle di "consulenza giuridica", mostra con chiarezza i limiti che l'informatica tradizionale incontra nel mondo del diritto:

(a) Un programma informatico tradizionale è limitato ad un tipo ben determinato di problemi (un programma per il calcolo della retribuzioni non potrebbe essere usato, ad esempio, per determinare posizioni previdenziali); nel mondo giuridico si presentano continuamente nuovi problemi, e i diversi settori del diritto debbono spesso essere correlati per determinare la disciplina da applicare ad un caso concreto.

(b) Ogni programma svolge un'operazione esattamente definita, sia nel suo obiettivo (il calcolo degli stipendi), sia nel suo contesto (la disciplina legislativa e negoziale in vigore in un momento dato); nei sistemi giuridici moderni è stato istituzionalizzato (e disciplinato giuridicamente) il continuo mutamento della normativa giuridica, così da consentire un rapido adeguamento del diritto alle esigenze sociali. Sono pertanto possibili continue modifiche normative che richiedono corrispondenti trasformazioni del programma, con costi e conseguenze spesso imprevedibili<sup>15</sup>.

<sup>14</sup>I diagrammi di flusso o diagrammi a blocchi sono un linguaggio grafico per rappresentare algoritmi.

<sup>15</sup>La crescente complessità dei programmi informatici causa la difficoltà di prevedere il comportamento dei programmi stessi, e di apprezzare l'impatto di modifiche successive. Questa complessità è alla base delle periodiche "crisi del software". I cambiamenti di paradigma dell'informatica che abbiamo illustrato nei capitoli precedenti possono

(c) L'elaborazione da compiere deve essere esattamente determinata in tutti i particolari, debbono essere esattamente individuati gli input del sistema, che corrisponderanno di regola ai dati da inserire in un determinato modello o formulario (o alle risposte ad una serie di domande predefinite proposte dall'elaboratore all'utente), e i corrispondenti output. Nel mondo del diritto, accanto a casi completamente standardizzati, si presentano casi nei quali elementi ulteriori realizzano un'eccezione disciplinata in modo diverso. Queste eccezioni possono non essere state contemplate nella progettazione del sistema, in quanto stabilite dalla normativa successiva o da normative settoriali.

(d) Raramente i documenti normativi adottano una formulazione di tipo algoritmico: il passaggio dai documenti originali (meglio, dall'interpretazione attribuita agli stessi) ad un linguaggio di tipo algoritmico comporta una radicale trasformazione. Occorre tradurre i contenuti giuridici nella composizione sequenziale (eventualmente strutturata) di operazioni elementari sui dati.

(e) Questa trasformazione comporta un allontanamento non solo dallo stile di un determinato documento normativo, ma, in generale, dal linguaggio nel quale gli uomini comunicano tra di loro. Ad essa si accompagna una perdita di trasparenza. Diventa estremamente difficile riconoscere, nel testo del programma, gli elementi che esprimono la conoscenza giuridica, da quelli destinati a regolare il processo computazionale. I due aspetti sono fusi in un unico programma, nelle stesse strutture formali.

(f) Il passaggio ad una formulazione di tipo algoritmico comporta inevitabilmente una perdita di astrazione. I programmi di questo tipo sono, come si suol dire deterministici: il programma deve determinare esattamente la sequenza delle operazioni da compiere sui dati. Non sempre, al fine di adempiere a prescrizioni giuridiche o per determinare delle qualificazioni giuridiche, è necessario osservare sequenze di comportamenti esattamente definite:

-Spesso la legge si limita ad indicare il comportamento al quale il cittadino è obbligato, senza specificare come questi debba procedere per determinare il comportamento cui è tenuto e porlo in essere (ad esempio, per determinare l'imposta dovuta, posso calcolare prima le detrazioni d'imposta, e poi l'imposta o viceversa; per adempiere all'obbligo posso prelevare la somma necessaria dal mio

---

anche essere interpretati come la risposta scientifico-tecnologica alle crisi del software. Per una considerazione del problema nell'informatica giuridica, cfr. FIEDLER H., BARTHEL T., VOOGD G., *Formalisierung im Recht. Untersuchungen zur formalisierung Recht als Beitrag zur Grundlagerecherche juristischer Datenverarbeitung*, Westdeutscher Verlag, Leverkusen, 1984.

conto in banca, chiederla in prestito ad amico, ottenerla vendendo un mio bene etc.). In questi casi, l'imposizione di una sequenza determinata è un'aggiunta non necessaria.

-Anche quando la legge stabilisca un procedimento, questo può comprendere atti necessari, atti eventuali, atti da compiersi in alternativa. Tra alcuni atti c'è un ordine determinato (in quanto il compimento degli uni condiziona la validità degli altri), tra altri non è definito alcun ordine. L'imposizione di una rigida sequenza è, anche in questi casi, una sovrapposizione artificiosa;

Una formalizzazione algoritmica più che una riformulazione del diritto è la precisazione di un metodo particolare per adempiere al diritto, o per determinare una qualificazione giuridica.

(g) Più in generale, per sviluppare un programma algoritmico occorre aver stabilito con precisione lo scopo che ci si ripromette di raggiungere, e l'utente cui il programma è destinato: scopo e tipo di utente concorrono nel determinare quali operazioni si debbano compiere in base al diritto. Il programma deve tenerne conto e diversi scopi o diversi utenti richiedono un diverso programma, seppure sviluppato a partire dalla medesima normativa. Ad esempio "un enunciato apparentemente semplice come "i richiedenti debbono presentare domanda entro il 1 gennaio", può essere letto in molti modi diversi. Esso indica a un richiedente quanto tempo egli abbia a disposizione per preparare la propria domanda. Aiuta l'amministrazione a determinare se una domanda dovrebbe essere respinta. Potrebbe essere usato da un giurista in argomentazioni circa il significato dei termini 'domanda valida' o di 'presentazione', così come usati in questa norma o in altre norme correlate". Ciascuna di queste letture comporta una diversa trascrizione algoritmica dell'enunciato di cui trattasi<sup>16</sup>.

I limiti che abbiamo elencato non toccano nello stesso modo tutte le applicazioni informatico-giuridiche. Allo stato della tecnologia, i metodi algoritmici (disciplinati nella programmazione strutturata) possono essere i più adatti per molte applicazioni standardizzate di tipo gestionale. Infatti questi metodi possono essere sufficienti per quelle applicazioni, mentre l'esigenza di gestire efficientemente grandi quantità di dati e la necessità di colloquiare con altri programmi informatici (in particolare, con i sistemi per la gestione di basi di dati) possono rendere preferibile o necessario il ricorso a strumenti consolidati. Invece, l'uso di tecniche dell'intelligenza artificiale nei sistemi di consultazione giuridica automatica sembra già oggi opportuno o necessario. Come

---

<sup>16</sup>SERGOT M.J., *The Representation of Law in Computer Programs: A Survey and Comparison of Past and Current Projects*, cit., p. 17 ss.

vedremo nel seguito, queste tecniche consentono di superare, almeno in parte, i limiti indicati nei punti da (a) a (f), (limiti particolarmente gravi per queste applicazioni), mentre le ragioni che suggeriscono l'uso di metodi consolidati non sono così forti per i sistemi di consulenza giuridica.

#### 2.1.4. La programmazione strutturata come modello del diritto

Nell'informatica giuridica si è dedicata una scarsa attenzione ai sistemi che realizzano forme di applicazione automatica della normativa giuridica in contesti gestionali. Piuttosto che sulle tecniche da usare per sviluppo di questi sistemi, l'attenzione del giurista si è concentrata sulla problematica del loro impatto sociale, in particolare in relazione alla tematica della tutela della riservatezza. Lo sviluppo delle applicazioni informatiche è stato affidato ad altri soggetti: come amministratori, analisti, programmatori, ecc.

Lo scarso interesse del giurista dipende probabilmente dal fatto che tali sistemi gli richiedono solo un input iniziale. La preparazione di un sistema di questo tipo presuppone una determinazione univoca del contesto normativo dello specifico, ristretto problema cui il sistema è destinato, ma la semplicità delle operazioni da automatizzare (prescindendo dalle operazioni di calcolo) rende inutile un ulteriore intervento del giurista. Inoltre, questi sistemi usano i risultati del lavoro giuridico (in senso stretto) piuttosto che sostituirsi allo stesso. Essi svolgono un'attività meramente applicativa che, benché abbia per oggetto norme giuridiche, non ha mai presupposto una specifica educazione giuridica (si pensi, ad esempio, alla redazione di certificati anagrafici, o al calcolo degli stipendi).

Benché le applicazioni gestionali abbiano toccato solo marginalmente il lavoro del giurista, esse hanno invece inciso profondamente sulle strutture organizzative pubbliche e private e sul rapporto tra cittadini e amministrazioni. L'applicazione (cieca e meccanica) di norme giuridiche già oggi è spesso affidata a sistemi automatici, e lo sarà in misura maggiore nel prossimo futuro. Per queste applicazioni il problema non è tanto decidere se sia possibile e lecita un'applicazione automatica del diritto, ma piuttosto esaminare se sia possibile sviluppare sistemi informatici più flessibili, trasparenti e controllabili, anche grazie alle tecnologie dell'intelligenza artificiale.

La realizzazione di programmi che applicano norme giuridiche ha ispirato, verso a fine degli anni 70, un interessante tentativo di trasferire il metodo della programmazione strutturata alla normativa giuridica. Si

parlava di una "legislazione strutturata", che riprendesse le strutture della programmazione strutturata, così da ottenere anche nel diritto i risultati di semplicità, di chiarezza, di comunicabilità che il metodo della programmazione strutturata intende conseguire<sup>17</sup>. Questo tentativo si è interrotto quando l'interesse delle ricerche informatico-giuridiche si è spostato verso le applicazioni di intelligenza artificiale.

I risultati ottenuti sono stati assai limitati: nonostante questa problematica abbia suscitato un notevole interesse, non vi sono state realizzazioni significative. Guardando retrospettivamente è facile rendersi conto delle ragioni di questo insuccesso. La programmazione tradizionale (algoritmica), non è il modo più naturale per interpretare la normativa giuridica poiché questa solo in rari casi è facilmente esprimibile come una combinazione, seppure strutturata, di operazioni su dati. La sostituzione di una rigida procedura, allo stile dichiarativo delle norme giuridiche (anche di quelle che prescrivono procedimenti) sembra del tutto inadeguata, tranne che in determinati contesti ristretti. Ciò non significa che in qualche caso non possa essere utile rappresentare con un diagramma di flusso il metodo per applicare una certa normativa giuridica (ad esempio, il calcolo delle imposte), o per realizzare un procedimento giuridico<sup>18</sup>. Quando ci si proponga di ottenere risultati più generali (e non immediatamente finalizzati alla redazione di un programma algoritmico) sembra preferibile ricorrere a metodi diversi e, in particolare, ai formalismi proposti dall'intelligenza artificiale.

Negli anni scorsi si è cercata un'alternativa ai diagrammi di flusso nelle reti di Petri, un formalismo di tipo reticolare, meno legato al funzionamento degli attuali

---

<sup>17</sup>Per una considerazione del rapporto tra programmazione strutturata e legislazione, cfr., in particolare, FIEDLER H., *Structured Programs as a Paradigm of Structured Laws*, in NIBLETT (a cura di), *Computer Science and Law*, cit, pp. 137-146; BARTHEL T., *Structured Programs as a Paradigm of Structured Laws*, in NIBLETT (a cura di), *Computer Science and Law*, 1979, pp. 147-156.

<sup>18</sup>Per alcuni esempi di applicazione del metodo della diagrammazione a blocchi a contesti giuridici cfr. LOSANO M., *Informatica per le scienze sociali*, cit., p. 430 ss. Un'interessante applicazione è riportata, in MERLO B., MICHELI G., *Il condono edilizio computerizzato*, Pirola, Milano, 1985, i quali ricorrono al metodo dei diagrammi di flusso per rappresentare le procedure per ottenere il condono edilizio. Per gran parte della legge Merlo e Micheli si limitano ad una rappresentazione schematica delle procedure giuridiche, per il calcolo della somma da pagare per ottenere la sanatoria propongono programmi interattivi in BASIC.



elaboratori (in particolare, consente di rappresentare processi concorrenti), più flessibile e vicino alla logica<sup>19</sup>. Anche le ricerche in questa direzione sembrano essersi oggi interrotte, poiché sembra preferibile passare direttamente a tecniche di rappresentazione della conoscenza giuridica ancor più evolute e lontane dalla programmazione tradizionale.

## 2.2. L'automazione d'ufficio come supporto dell'attività del giurista

Il lavoro del giurista si svolge sempre più nel contesto delle tecniche per l'automazione del lavoro d'ufficio, la c.d. buroatica.

Sempre più numerosi studi professionali usano strumenti informatici per l'elaborazione di testi, per la tenuta di archivi relativi ai clienti (e, in generale, alle attività in corso), la gestione della contabilità, la redazione di documenti standardizzati. I medesimi strumenti sono largamente usati nella pubblica amministrazione e nell'amministrazione della giustizia.

Finora l'automazione d'ufficio ha toccato solo indirettamente il lavoro giuridico in senso stretto pur modificando profondamente il contesto nel quale quel lavoro si svolge (ad esempio aumentando l'efficienza e la produttività dell'ufficio). Infatti, queste applicazioni per lo più riguardano direttamente il lavoro di taluni collaboratori del giurista (dattilogafi, segretari ecc.), lavoro che non richiede specifiche competenze giuridiche.

Negli ultimi anni, l'intelligenza artificiale è stata applicata anche all'automazione del lavoro d'ufficio. Si stanno realizzando sistemi automatici in grado di scrivere sotto dettatura, di correggere errori sintattici, di redigere documenti complessi. Tuttavia, non sembra che le tecniche per la gestione d'ufficio potranno modificare in misura significativa il lavoro giuridico in senso stretto. La specificità dei problemi giuridici rende difficilmente concepibile un'informatizzazione mediante strumenti progettati in generale per la gestione d'ufficio.

Vi sono alcuni programmi per la gestione d'ufficio specificamente destinati ad un'utenza giuridica. Un aspetto interessante di tali programmi è la possibilità di una

---

<sup>19</sup>Sulle reti di PETRI nel diritto, cfr. DEGLI ANTONI G., ZONTA B., *Analysis of Laws by Means of Petri Nets: Motivations and Metodolology*, in CIAMPI (a cura di) *Artificial Intelligence and Legal Information Systems*, 1982, pp. 273-299; SARTOR G., *Le reti di Petri. Un linguaggio formale applicabile al diritto?*, in "Sociologia del diritto", 1986, pp. 109-118, anche per alcune indicazioni bibliografiche.

redazione parzialmente automatica di alcuni tipi di atti giuridici standardizzati (funzione particolarmente importante negli studi notarili). Poiché la redazione degli atti giuridici è disciplinata dal diritto, l'applicazione automatica di norme giuridiche viene spesso integrata nella redazione dei testi. Il giurista può procedere alla redazione dell'atto standardizzato (di alcune parti di un atto) limitandosi a fornire le parti variabili a seconda del caso. Anche programmi per la redazione automatica di atti giuridici presentano limiti che si connettono all'impiego di tecniche informatiche tradizionali. Detti limiti che si manifestano soprattutto nella rigidità del sistema, che impedisce al giurista di redigere gli atti in base alle proprie scelte personali.

### 2.3. Le basi di dati giuridiche

Le basi di dati giuridiche sono la realizzazione più significativa dell'informatica giuridica, a tal punto che la progettazione la gestione e l'uso di banche dati giuridici spesso si identificano per il profano con l'informatica giuridica<sup>20</sup>.

Si tratta di sistemi di documentazione automatica che contengono una varia tipologia di testi giuridicamente rilevanti. Ad esempio, nel sistema Italgiure, gestito presso la Corte di Cassazione italiana, sono compresi atti legislativi nazionali e regionali (in testo integrale), la giurisprudenza della Corte Costituzionale, della Corte di Cassazione e una parte della giurisprudenza di merito (nella forma di massime), testi di dottrina giuridica (titoli ed estratti di libri e articoli) e di informatica giuridica ecc.

I sistemi per la gestione di basi di dati giuridici si propongono di fornire all'utente i documenti contenuti nella base di dati che attengono all'argomento al quale egli è interessato. I metodi per il reperimento dei documenti sono

---

<sup>20</sup>Questo fenomeno si riconnette non solo all'importanza delle basi di dati giuridici, ma anche a taluni problemi terminologici. Infatti si propone di usare il neologismo informatica per designare specificamente "la disciplina che si occupa della ricerca documentaria mediante l'elaboratore elettronico" (LOSANO M., *Informatica per le scienze sociali*, cit., p. 337 ss.), anziché lo studio dell'elaborazione automatica di informazioni in generale. Peraltro, l'accezione più generale del termine si è ormai imposta.

Un altro problema terminologico attiene ai concetti di banca dati e base di dati: sono state proposte numerose definizioni, spesso incompatibili tra loro, che distinguono tra i due concetti. Per evitare tali problemi definitori, userò i due termini come sinonimi.

molteplici, di regola è possibile accedere a tutti i documenti che contengano determinate parole (utilizzando eventualmente il troncamento o forme di lemmatizzazione<sup>21</sup>), o che siano stati associati a determinati descrittori o parole chiave<sup>22</sup>, eventualmente strutturati in un thesaurus<sup>23</sup>.

Le banche di dati giuridici costituiscono già oggi una importante realtà, anche se il loro uso è relativamente limitato e, di conseguenza, la loro gestione, con qualche rara eccezione, non si è rivelata economicamente vantaggiosa. Pertanto, in tutti gli Stati dell'Europa occidentale, fino a poco tempo fa, si manifestava la tendenza verso la concentrazione della documentazione giuridica automatica in un unico grande sistema, gestito con l'aiuto di consistenti contributi pubblici<sup>24</sup>. Recentemente, grazie ad alcune innovazioni tecnologiche, si sono verificate interessanti novità nel mercato delle banche di dati giuridici:

(a) L'informatizzazione dell'editoria ha condotto alla realizzazione di banche dati che rappresentano un sottoprodotto dell'attività editoriale. La memorizzazione dei testi su supporto magnetico effettuata ai fini della pubblicazione può essere usata anche per la realizzazione di una base di dati, eliminando così i costi dell'immissione dei dati<sup>25</sup>.

(b) La tecnologia dei dischi ottici ha consentito di realizzare a basso prezzo archivi di notevoli dimensioni

---

<sup>21</sup>La ricerca mediante troncamento consente di reperire tutti i testi che contengono vocaboli che iniziano in un certo modo; la ricerca mediante lemma consente di ottenere i testi che contengono parole riconducibili ad una certa radice.

<sup>22</sup>Ad esempio nel già menzionato sistema Italgiure, le massime della Corte di Cassazione possono essere reperiti utilizzando una pluralità di metodi o "canali" di ricerca, come i seguenti: (a) le parole contenute nel testo, (b) i lemmi cui quelle parole sono riconducibili, (c) il testo di legge citato (d) la data ecc. Sul sistema Italgiure, cfr. NOVELLI V., GIANNANTONIO E., *Manuale per la ricerca elettronica dei documenti giuridici*, Giuffrè, Milano, 1982.

<sup>23</sup>Sui thesauri giuridici, mi permetto di rinviare a CIAMPI C., SARTOR G., *Un thesaurus di descrittori per la legge sul condono edilizio*, in Quaderni dell'ISAS, 1986, pp. 1-22.

<sup>24</sup>Sull'uso delle banche di dati giuridici in Europa, cfr. LLOYD M., *Legal Databases in Europe. User Attitudes and Supplier Strategies*, North Holland, Amsterdam, 1986.

<sup>25</sup>Ad esempio, la banca di dati GURITEL, del Poligrafico dello Stato contiene tutti i testi pubblicati nella Gazzetta ufficiale.

gestibili su microelaboratore. Negli ultimi anni numerose banche di dati giuridici su disco ottico sono state realizzate e commercializzate da parte di imprese private (di solito si tratta di case editrici). Le basi di dati su disco ottico non possono competere con i grandi sistemi telematici per quanto riguarda la completezza e la rapidità di aggiornamento: ciascuna di esse comprende un settore determinato della normativa o della giurisprudenza, e viene aggiornata con una certa periodicità (di regola annuale o semestrale). Il loro successo si spiega soprattutto con la semplicità e l'economicità di utilizzo<sup>26</sup>.

Sono prevedibili nel prossimo futuro notevoli sviluppi tecnologici nel campo delle basi di dati giuridici:

- Nuovi metodi di interrogazione e di ricerca. Le tecniche dell'intelligenza artificiale per il trattamento del linguaggio potranno consentire una più agevole interazione con il sistema e il reperimento dei testi anche sulla base (della ricostruzione automatica) del loro significato.

- Economicità e facilità di accesso. Il perfezionamento delle telecomunicazioni e delle interfacce potrà semplificare l'accesso alle banche dati telematiche. Lo sviluppo della tecnologia dei dischi ottici potrà consentire la costituzione e l'uso su microelaboratore di basi di dati locali che avranno poco da invidiare ai grandi sistemi telematici.

Anche se nuove tecnologie daranno dimensioni maggiori al fenomeno delle banche dati giuridiche, bisogna sottolineare il limite di questo importante strumento del giurista, limite che, peraltro, ne rappresenta il pregio e lo rende insostituibile. I sistemi di questo tipo si limitano a fornire un sottoinsieme dei testi memorizzati nell'archivio, senza compiere alcuna elaborazione sugli stessi. Spesso il giurista è interessato solo al reperimento di testi giuridici che rispondano a determinati requisiti: ad esempio, i testi attinenti ad un determinato argomento e/o di una categoria determinata e/o pubblicati in un determinato intervallo temporale ecc. In tutti questi casi un sistema di documentazione automatica è lo strumento necessario e sufficiente per ottenere il risultato desiderato.

Quindi, i sistemi di documentazione automatica svolgono una funzione importantissima, ma limitata rispetto al lavoro del giurista: essi contribuiscono alla sola attività di ricerca documentaria e, in alcuni casi, solo ad una parte di essa. Infatti se nella base di dati i testi cercati sono

---

<sup>26</sup>L'uso di dischi ottici a fini di documentazione giuridica ha conosciuto una straordinaria espansione negli ultimi anni. Cfr. LOSANO M., *Scritto con la luce*, Unicopli, Milano, 1988.

memorizzati per intero, l'utente potrà accedere direttamente ad essi (ad esempio, ottenere una stampa del testo legislativo al quale è interessato). Altre volte l'utente otterrà un'informazione parziale da integrare se necessario con una ricerca tradizionale (ad esempio, se non ritenga sufficiente la massima di una sentenza dovrà cercare il testo integrale nel repertorio). Altre volte, infine, il sistema fornirà solo indicazioni per localizzare i testi rilevanti (ad esempio, la data e il numero di pubblicazione di una legge, o il titolo e l'autore di un'opera di dottrina ecc.).

La ricerca documentaria rientra per taluni aspetti nel lavoro giuridico in senso stretto: essa richiede spesso una cultura giuridica (la conoscenza delle fonti del diritto, della terminologia, degli istituti, ecc.). E' da osservare tuttavia che, quanto più si perfezioneranno le interfacce per l'interrogazioni delle banche dati e le tecniche di ricerca, tanto più sarà facilitata l'attività di ricerca documentaria, e diverrà meno necessaria una cultura giuridica approfondita, almeno per ricerche standardizzate<sup>27</sup>.

#### **2.4. Conclusione: Limiti dell'uso degli strumenti informatici tradizionali nel diritto**

In conclusione, si può affermare che le forme di automazione che corrispondono al paradigma della "gestione di dati", toccano solo marginalmente l'attività giuridica in senso stretto, in quanto limitate a:

(a) l'individuazione e all'applicazione di norme giuridiche in settori estremamente circoscritti e sul presupposto di una completa specificazione delle operazioni da compiere per applicare quelle norme. Si tratta solamente di operazioni esecutive, estranee al lavoro giuridico in senso stretto;

(b) l'attività (burotica) di mero supporto esterno rispetto a tale lavoro;

(c) la ricerca documentaria, il primo e più semplice aspetto del lavoro del giurista.

---

<sup>27</sup>I prevedibili perfezionamenti dei sistemi per la gestione di basi di dati giuridiche vanno dalla creazione di interfacce più vicine all'utenti (*user friendly*), alla possibilità di sfruttare meglio il significato dei termini della ricerca (con strumenti come i *thesauri*), fino allo sviluppo di sistemi di reperimento di informazioni concettuale (*conceptual information retrieval systems*) cui si accennerà nell'introduzione al cap. 4.

### 3. I comportamenti intelligenti

Come si è accennato in precedenza, le realizzazioni informatico-giuridiche operative ancor oggi corrispondono tutte al tipo di sistemi presentati nelle pagine precedenti, ma negli anni 80 l'interesse della ricerca si è spostato da verso l'intelligenza artificiale, e, in particolare, verso la realizzazione di sistemi basati sulla conoscenza giuridica.

Questa problematica sarà trattata ampiamente nei cap. 3 e 4. Qui mi basta sottolineare due aspetti:

-Il sostanziale fallimento dell'informatizzazione dell'attività giuridica in senso stretto mediante tecniche informatiche tradizionali, induce a ricorrere alle tecnologie dell'intelligenza artificiale (sulle quali si riversano talvolta le aspettative e i timori che accompagnarono il primo ingresso dell'informatica nel mondo del diritto)<sup>28</sup>.

-Il tentativo di realizzare applicazioni giuridiche dell'intelligenza artificiale solleva questioni vicine a classici temi della scienza e della teoria del diritto. Per informatizzare il lavoro giuridico in senso stretto non basta più il trasferimento nel mondo del diritto di tecnologie applicabili e applicate in altri settori<sup>29</sup>, è necessario uno stretto rapporto tra studi informatici e studi giuridici (e, in particolare filosofico-giuridici)<sup>30</sup>.

Nel paragrafo precedente si è visto che l'intelligenza artificiale si ispira fondamentalmente al modello del sistema deduttivo, e da ciò si sono desunti taluni limiti di tali applicazioni. Nelle pagine seguenti cercheremo di considerare brevemente in quale misura tali limiti

---

<sup>28</sup>Come è noto, agli inizi dell'informatica giuridica molti si attendevano un rapido e radicale sconvolgimento dell'attività del giurista da parte dell'informatica. Queste previsioni non si sono affatto realizzate.

<sup>29</sup>Ad esempio, gli attuali sistemi per la gestione di basi di dati giuridici non differiscono sostanzialmente dai sistemi di documentazione automatica introdotti ad esempio, nel settore della documentazione scientifica, delle biblioteche, ecc. Nel campo della ricerca documentaria, solo lo sviluppo di sistemi per il reperimento di informazioni concettuale, che tengano conto del significato dei testi memorizzati, sembra richiedere un'approfondita analisi filosofico-giuridica.

<sup>30</sup>Il tema del rapporto tra gli studi di teoria e filosofia del diritto, e le applicazioni giuridiche dell'intelligenza artificiale, sarà affrontato nel cap. 6, par. 8..

riguardano anche le applicazioni di intelligenza artificiale nel diritto. Ci limiteremo ad alcune considerazioni sommarie, che cercheremo di approfondire nel seguito. Dato lo stato iniziale delle ricerche in materia, si tratta solo di ipotesi che saranno verificate dai futuri sviluppi dell'informatica giuridica.

### 3.1. Il problema dell'esplosione combinatoria nel diritto

Il problema dell'esplosione combinatoria non sembra costituire un insormontabile limite allo sviluppo di sistemi basati sulla conoscenza giuridica giuridici.

In particolare, lo sviluppo di sistemi di analisi giuridica, cioè di sistemi con la funzione di determinare la qualificazione giuridica di una situazione data<sup>31</sup>, non richiede necessariamente l'uso di metodi euristici che pregiudichino la completezza della ricerca.

Il problema dell'esplosione combinatoria sorge di regola quando l'inferenza (o il cammino) che conduce alla soluzione del problema desiderata si componga di una sequenza di passi di una certa lunghezza, e in ogni passo si debba scegliere tra più assiomi e/o regole di inferenza, ciascuno dei quali sia ugualmente applicabile. Si tratta di una situazione che si realizza facilmente nei giochi e nei problemi logico-matematici<sup>32</sup>.

La deduzione di un "teorema giuridico", cioè della disciplina giuridica di una fattispecie determinata non sembra presentare problema di questo tipo. Il processo di inferenza per dedurre la disciplina della fattispecie è, di regola, relativamente semplice, qualora le norme da applicare siano già state rappresentate in un linguaggio formale in modo che sia possibile la loro applicazione automatica, e si rappresenti nello stesso linguaggio anche la fattispecie. Infatti, l'ordinamento giuridico, o anche solo un settore di esso, si compone di un numero enorme di regole, ma rispetto alla descrizione di una situazione determinata, solo alcune di esse trovano applicazione. Il problema da risolvere è soprattutto quello del rapido esame delle regole (più in generale, delle strutture

---

<sup>31</sup>Cfr. introduzione al cap. 4.

<sup>32</sup>Ad esempio, nel gioco degli scacchi, il giocatore che inizia la partita ha 10 possibili prime mosse, altrettante sono le possibili prime mosse dell'avversario, supponiamo che siano 10 le possibili seconde mosse di entrambi i giocatori, allora, in sole 4 mosse, abbiamo circa 10.000 possibili stati da considerare. Gli situazioni da considerare in una partita intera sono 10 alla 120, una cifra che non riusciamo neppure ad concepire.

rappresentative) contenute nella base di conoscenza per determinare quale regola sia applicabile alla fattispecie (compito nel quale un sistema automatico è enormemente efficiente), piuttosto che quello della scelta tra più regole tutte egualmente applicabili (dove le capacità di giudizio diventano determinanti, e in contesti complessi, insostituibili).

L'esplosione combinatoria può rappresentare invece un serio problema nei sistemi di pianificazione giuridica<sup>33</sup>, che si propongono di suggerire la linea di condotta più opportuna per raggiungere un certo scopo giuridico. Peraltro, in questi casi sembra più accettabile l'uso di metodi euristici<sup>34</sup>.

### 3.2. Il problema della formalizzazione nel diritto (rinvio)

Il problema della formalizzazione ha invece un'importanza centrale nella realizzazione di applicazioni giuridiche dell'intelligenza artificiale. Proprio poiché le questioni giuridiche, una volta ridotte alla dimensione del ragionamento deduttivo diventano relativamente semplici, il problema principale consiste nella costruzione di un contesto deduttivo dal quale ci si possa attendere la soluzione di concreti problemi giuridici. L'uso dell'intelligenza artificiale nel diritto riposa sulla possibilità di formalizzare contesti giuridici (come si vedrà nel seguito, consideriamo i concetti di "formalizzazione" e di "produzione di un sistema assiomatico formale" come tendenzialmente equivalenti).

L'ampia diffusione di programmi tradizionali che realizzano un'applicazione automatica di norme giuridiche, prova che in una certa misura la formalizzazione del diritto è possibile. Infatti, questi sistemi si basano su una rappresentazione formale del processo di applicazione delle norme, e quindi su una implicita formalizzazione delle norme stesse. L'intelligenza artificiale ripropone il problema in una nuova dimensione, in quanto (a) affronta direttamente il problema della rappresentazione formale della normativa giuridica e (b) intende affrontare anche problemi complessi, attinenti al lavoro giuridico in senso stretto.

Nel pensiero filosofico-giuridico l'uso di strumenti deduttivi ha trovato numerose critiche.

---

<sup>33</sup>Cfr. introduzione al cap. 4.

<sup>34</sup>L'utente che chieda, ad esempio, qual'è l'assetto patrimoniale che gli consenta di minimizzare l'imposizione fiscale, si attende una risposta soddisfacente, non necessariamente l'unica o la migliore possibile.



Taluni autori hanno ridimensionato l'importanza del ragionamento deduttivo, negando che esso costituisca l'aspetto esclusivo o preminente nell'attività del giurista: prima di applicare dei metodi deduttivi il giurista deve identificare le premesse da cui muovere. A tal fine è necessario assegnare un significato determinato ai documenti normativi e qualificare in un determinato modo i fatti del caso concreto, operazioni che fondano la possibilità di una applicazione del metodo deduttivo, ma sfuggono allo stesso.

Altri sono andati più in là, affermando una radicale incompatibilità tra metodo deduttivo e ragionamento giuridico. Mi limito a menzionare due posizioni particolarmente significative:

-La cosiddetta nuova retorica, contrappone al ragionamento logico-matematico, la retorica o l'argomentazione come specifiche logiche dei contesti morali e giuridici<sup>35</sup>.

-Si è speso argomentata l'impossibilità di applicare metodi deduttivi nel diritto dalla natura prescrittiva delle norme giuridiche, che impedisce l'assegnazione di un valore di verità alle stesse<sup>36</sup>.

Qui importa sottolineare che una risposta affermativa al problema della possibilità e dell'utilità di un'applicazione di metodi deduttivi, e della loro premessa insostituibile, la formalizzazione, (intesa soprattutto come rappresentazione del diritto in un sistema assiomatico formale) ha un valore pregiudiziale rispetto allo sviluppo di sistemi esperti giuridici. Se quei metodi e quel tipo di rappresentazione non possono essere utilizzati in contesti giuridici, allora lo sviluppo di sistemi esperti giuridici, cioè di sistemi basati sostanzialmente sul paradigma del sistema deduttivo è impossibile o inutile.

Nel configurare la formalizzazione come premessa necessaria del funzionamento di un sistema informatico intelligente, sembra si sia escluso che il sistema possa sostituire il giurista nell'aspetto più importante e peculiare del lavoro del giurista, l'interpretazione. La rappresentazione del diritto quale sistema deduttivo dal

---

<sup>35</sup>Cfr, tra gli autori riconducibili a questo indirizzo PERELMAN Ch., OLBRECHTS-TYTECA L., *La nouvelle rethorique. Traité de l'argumentation*, Presses Universitaires de France, Paris 1958; VIEHWEG T., *Topik und Jurisprudenz. Ein Beitrag zur rechtswissenschaftlichen Grundlagenforschung*, C.H. Beck'sche Verlagsbuchhandlung, München, 1965 (prima ed. 1954).

<sup>36</sup>Su questa problematica, anche per alcune indicazioni bibliografiche, mi permetto di rinviare a SARTOR G., *Alcune osservazioni sull'applicabilità della logica classica al diritto*, in "Informatica e diritto", 3, 1987, pp. 65-90.

quale discenda la soluzione di un caso determinato non è il presupposto, ma il risultato dell'attività del giurista. E' il risultato dell'ascrizione di un determinato significato a dei testi in linguaggio naturale e della qualificazione in un determinato modo dei fatti da valutare (qualificazione da esprimersi nel medesimo linguaggio nel quale si esprime il significato ascritto ai testi normativi). La rappresentazione del diritto come sistema assiomatico formale presuppone inoltre che sia l'interpretazione del testo normativo, sia la qualificazione dei fatti siano espresse in un linguaggio formale<sup>37</sup>.

Se un sistema informatico si limita a compiere operazioni deduttive, e queste, nel diritto, hanno un'importanza assai limitata, sembra se ne debba dedurre che anche le tecniche dell'intelligenza artificiale siano destinate ad avere uno scarso impatto sul mondo del diritto.

Come vedremo nel seguito, questa conclusione è accettabile solo in parte. Invero:

(a) Una volta che un certo contesto normativo sia stato formalizzato, diventa possibile usare questa formalizzazione anche per risolvere un numero illimitato di casi futuri standardizzati, senza ripetere l'analisi che ha portato a quella determinata formalizzazione: il sistema automatico può limitarsi a compiere le operazioni deduttive per le quali è stato predisposto.

(b) Strumenti automatici possono agevolare la preparazione di un modello deduttivo. Si tratta di strumenti per il trattamento del linguaggio naturale, per lo sviluppo di ragionamenti analogici ecc.

(c) Si può auspicare che l'uso di tecniche informatiche nella redazione dei testi normativi conduca ad una legislazione più semplice e razionale, priva almeno in parte delle incertezze di significato che caratterizzano il diritto vigente. Una legislazione adatta all'automazione, dovrebbe facilitare il passaggio dal testo legislativo alla sua rappresentazione formale<sup>38</sup>.

Affronteremo gli aspetti appena menzionati nel seguito del nostro volume. Mi limito ad osservare che tutte le prospettive appena delineate sollevano delicati problemi:

(a) L'applicazione di una formalizzazione astratta a nuovi casi impone controlli e cautele per evitare che la

---

<sup>37</sup>Il rapporto tra interpretazione e sviluppo di applicazioni di intelligenza artificiale sarà discusso nel cap. 6.

<sup>38</sup>Da tempo si parla della necessità di sviluppare una tecnica legislativa adatta all'informatica o all'automazione. Come è noto, già nel 1972, il Land tedesco della Baviera emanava una legge che stabiliva i principi di una *automationsgerechte Gesetzgebung*.

cieca deduzione delle conseguenze di tale formalizzazione conduca a risultati aberranti.

(b) I metodi per l'analisi del linguaggio naturale e per il trattamento del ragionamento analogico non si sono rivelati concretamente utilizzabili in nessun sistema informatico-giuridico operativo. In prospettiva, quei metodi, considerati gli aspetti valutativi insiti nell'attività giuridica, possono consentire di realizzare sistemi che siano solo ausili del giurista e non suoi sostituti;

(c) L'esigenza di adeguare il linguaggio giuridico all'informatica non deve far dimenticare che sono i fruitori di quel linguaggio, e quali sono i problemi sociali che la normativa giuridica deve risolvere:

-Chiarezza per l'uomo e chiarezza per l'elaboratore non sempre coincidono. Un testo in linguaggio di programmazione, benché eseguibile automaticamente può risultare del tutto incomprensibile, anche per un programmatore esperto. I programmi di una certa dimensione tendono ad assumere una complessità tale che nessuno riesce a dominarli<sup>39</sup>.

-Proprio l'indeterminatezza di significato del linguaggio naturale lo rende strumento idoneo a regolare situazioni future e non del tutto prevedibili.

-Ogni linguaggio formale è dotato di una capacità espressiva limitata rispetto al linguaggio naturale. Sarebbe grave che la scelta di adottare un linguaggio formalizzato (o comunque ridotto) predeterminasse non solo la forma ma anche i possibili contenuti e scopi della disciplina giuridica.

---

<sup>39</sup>I problemi della comprensibilità, della chiarezza, del coordinamento delle strutture linguistiche che rappresentano contenuti complessi si presentano sia nel diritto che nell'informatica. Paradossalmente, mentre i giuristi si rivolgono all'informatica per cercare il rimedio ai problemi della complessità e dell'oscurità delle leggi, gli informatici si rivolgono alle scienze umane, e, in particolare, al diritto, nella ricerca di metodi e linguaggi per trattare conoscenze complesse.



## CAP. 3

# I SISTEMI BASATI SULLA CONOSCENZA

L'intelligenza artificiale si propone di realizzare strumenti informatici in grado di compiere attività che, se fossero svolte da uomini, sarebbero attribuite alla loro intelligenza. Lo svolgimento di attività intelligenti si basa sull'impiego attivo, non rigidamente predeterminato, della conoscenza. Di qui l'esigenza di sviluppare un nuovo tipo di sistemi informatici, i cosiddetti sistemi basati sulla conoscenza (*knowledge based systems*), i quali si propongono di usare in modo intelligente le informazioni, di trasformare dati in conoscenza. Seguendo lo schema già usato nei capitoli precedenti, nel presente capitolo cercheremo di esaminare le principali caratteristiche di questi nuovi sistemi informatici; nel capitolo seguente approfondiremo l'analisi delle realizzazioni e delle prospettive nel mondo del diritto.

### 1. I sistemi di conoscenza

Anche i programmi informatici tradizionali incorporano conoscenza (ad esempio, un sistema per il calcolo delle retribuzioni si basa sulla normativa fiscale, sulla disciplina legislativa e contrattuale dei rapporti di lavoro ecc.). Infatti essi non sono che la descrizione o la prescrizione della procedura o dell'algoritmo per svolgere un certo compito e possono essere sviluppati solo tenendo conto delle caratteristiche di quel compito, ma:

- La conoscenza non è rappresentata esplicitamente, non è separata dalle procedure che disciplinano il processo di elaborazione, che usano la conoscenza.

- La conoscenza è applicata in modo rigidamente predeterminato (mediante l'esecuzione di tali procedure).

- Non è possibile aggiungere nuova conoscenza senza modificare le procedure.

- Il sistema non è in grado né di esporre la conoscenza sulla quale si basa, né di spiegare perché, sulla base di tale conoscenza, sia giunto a determinati risultati.

Invece, nei sistemi di conoscenza, di regola:

- La conoscenza è contenuta in una base di conoscenza (*knowledge base*), dove è rappresentata utilizzando un linguaggio ad alto livello, cioè un formalismo relativamente vicino ai linguaggi usati nella comunicazione umana. E' possibile adottare una rappresentazione dichiarativa del compito che il sistema deve svolgere, lasciando al sistema stesso l'individuazione della procedura da seguire. I

sistemi di conoscenza rappresentano lo sviluppo di un processo già iniziato con le basi di dati, processo tendente a rendere le informazioni usate da una certa applicazione informatica, relativamente indipendenti dai programmi che operano su di esse, a separare la logica dal controllo.

-La conoscenza è usata da un motore inferenziale, cioè da un programma in grado di compiere operazioni inferenziali, di effettuare deduzioni usando la base di conoscenza.

-E' possibile arricchire la base di conoscenza di nuove informazioni senza modificare il motore inferenziale.

-Il sistema è in grado di esporre in forma comprensibile le premesse e le argomentazioni che hanno condotto ad un determinato risultato, cioè di giustificare le conclusioni cui giunge<sup>1</sup>.

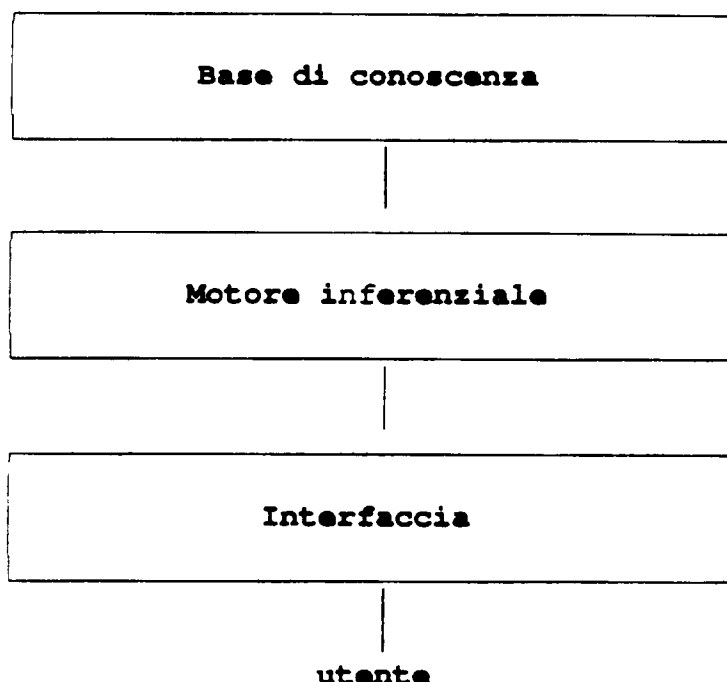
I sistemi basati sulla conoscenza sono spesso completati da interfacce che agevolano l'interrogazione e la preparazione della base della conoscenza. Talvolta l'utente può usare un sottoinsieme del linguaggio naturale nell'interazione con il sistema.

La struttura di un sistema basato sulla conoscenza è riportata nella fig. 1.

---

<sup>1</sup>Per illustrare la diversità tra l'uso delle informazioni nei programmi tradizionali e nei sistemi basati sulla conoscenza, SOWA J.F., *Conceptual Structures*, cit., 1984, p. 277-278, richiama le seguenti parole di SENECA L. A., *Ad Lucilium epistularum moralium libri* (trad. it. *Lettere a Lucilio*, Rizzoli, Milano, 1987), 33: "Ricordare è custodire ciò che è stato affidato alla memoria, mentre sapere significa far proprie le nozioni apprese, e non star sempre attaccato al modello, con lo sguardo sempre rivolto al maestro" e le commenta come segue "I programmi tradizionali incorporano la concezione di SENECA di memoria meccanica (*rote memory*): essi seguono ciecamente un insieme determinato di istruzioni e non sono in grado di adattare ai mutamenti delle circostanze. I sistemi basati sulla conoscenza, invece, incorporano conoscenza che acquistano da esperti umani, ma la applicano in modi nuovi a circostanze diverse".

**Fig. 1. Schema di sistema basato sulla conoscenza**



## **2. I sistemi esperti**

Un sistema basato sulla conoscenza è chiamato *sistema esperto* quando sia in grado di eseguire compiti che richiedono conoscenza specializzata, che possono essere svolti solo da esperti o comunque da persone dotate di notevoli competenze. Il concetto di sistema esperto risulta quindi da due elementi:

- Un elemento strutturale: deve trattarsi di un sistema basato sulla conoscenza e quindi, in particolare, composto di una base di conoscenza distinta da un motore inferenziale.

- Un elemento funzionale: il sistema deve essere in grado di fornire prestazioni che richiedano notevoli competenze<sup>2</sup>.

---

<sup>2</sup>Ogni contributo sui sistemi esperti contiene una diversa definizione del concetto di sistema esperto, così come ogni contributo sui sistemi esperti giuridici contiene una diversa definizione del concetto di sistema esperto giuridico. Pertanto, c'è il pericolo non solo della confusione, ma anche dello sviluppo di discussioni terminologiche di dubbia utilità. Ciò si riconnette al fatto che il termine "sistema esperto" ha assunto una connotazione positiva, e i sistemi che aspirano ad essere qualificati come "esperti" sono sempre più numerosi. Pertanto,

### 3. Sviluppo della tecnologia dei sistemi di conoscenza

Nello sviluppo delle applicazioni di intelligenza artificiale negli Stati Uniti d'America si sono distinti schematicamente tre momenti: una tesi, un'antitesi e una sintesi<sup>3</sup>.

#### 3.1. La tesi

La tesi che dominò una prima fase dell'intelligenza artificiale fu la seguente: possono essere trovati metodi generali per la soluzione di problemi, essi possono essere resi computazionali (utilizzabili da un elaboratore) e applicati a molti problemi diversi. Questo approccio si basa quindi sull'idea che il metodo o la strategia per risolvere un problema (la conoscenza procedurale, relativa al modo nel quale un compito può essere svolto efficientemente) sia indipendente dal particolare problema di cui si tratta. Basta fornire al sistema automatico l'esatta descrizione del compito e del contesto (la conoscenza dichiarativa, relativa alla natura del compito) affinché il sistema, applicando un metodo generale possa trovare la soluzione.

La ricerca di un metodo generale conduceva nella direzione della logica simbolica. Questo approccio ispirò la realizzazione di alcuni sistemi, soprattutto nel campo della dimostrazione automatica di teoremi. In particolare, bisogna ricordare due programmi sviluppati da Newell e Simon, che ebbero un'importanza fondamentale nella storia dell'intelligenza artificiale: il teoreta logico (*Logical Theorist*) e il risolutore di problemi generale (*General Problem Solver*)<sup>4</sup>.

---

<sup>3</sup>Per questa ricostruzione, cfr. WALKER A., *Knowledge systems: Principles and Practice*, in "IBM Journal of Research and Development", 1986, p. 3 ss.; WALKER A., *Knowledge systems: Principles and Practice*, in WALKER A., MCCORD M., SOWA J.F., WILSON W.G., *Expert Systems and Prolog*, Addison-Wesley, Reading (Massachusetts), 1987, p. 5 ss.

<sup>4</sup>Cfr. NEWELL A., SHAW J.C., SIMON H.A., *Empirical Exploration with the Logic Theory Machine: a Case Study in Heuristics*, in FEIGENBAUM E.A., FELDMAN J. (a cura di) *Computers and Thought*, cit., pp. 109-133; NEWELL A., SIMON H.A., *GPS, a Program that Simulates Human Thought*, in FEIGENBAUM E.A., FELDMAN J. (a cura di) *Computers and thought*, Mc Graw-Hill, New York, 1963, pp. 279-293.



### 3.2. L'antitesi

La tesi appena illustrata si scontrò con il problema dell'esplosione combinatoria<sup>5</sup>, che sembrò precludere la realizzazione di sistemi di applicazione generale abbastanza efficienti da essere concretamente utilizzabili.

Di qui il nascere di un'antitesi: anziché cercare di raggiungere ad ogni costo la generalità, bisogna individuare, per ogni tipo di problema, le tecniche più adatte per la sua soluzione, la specifica conoscenza procedurale necessaria, e incorporarla (o "implementarla", come si dice nel gergo informatico) nel sistema destinato ad risolverle quel tipo di problema. E' necessario allontanarsi dalle regole di inferenza della logica, la cui eccessiva generalità impedisce un'efficienza sufficiente nei diversi contesti. Ogni compito richiede invece, in linea di principio, metodi specifici e quindi un diverso programma.

Il programma che meglio rappresenta questo approccio è forse DENDRAL, che fungeva da supporto per analisi chimiche. In particolare, DENDRAL poteva essere usato per determinare la struttura di composti organici. L'uso di questo programma consentì di realizzare alcune scoperte significative<sup>6</sup>.

Questo approccio di tipo pragmatico condusse alla realizzazione dei primi sistemi basati sulla conoscenza suscettibili di utilizzazione commerciale. Esso, in cambio di un aumento di efficienza, comportava una perdita di generalità e la necessità di un'enorme quantità di lavoro intellettuale: la realizzazione di motori inferenziali (cioè di programmi difficili e complessi) diversi a seconda del tipo di problema postulava una lunga e difficile collaborazione tra esperto, ingegnere della conoscenza e programmatore.

### 3.3. La sintesi

Verso la metà degli anni 70, una sintesi tra tesi ed antitesi fu trovata nella realizzazione dei cosiddetti gusci (*shells*). Anche non è possibile sviluppare sistemi di impiego universale, è possibile sviluppare sistemi che rispondano ai requisiti comuni ad un certo insieme di compiti. A questi sistemi ai quali può essere poi essere aggiunta la conoscenza specializzata relativa ad un compito determinato. In particolare, è importante che la conoscenza aggiuntiva, relativa ad ogni compito particolare, sia solo o

---

<sup>5</sup>Cfr. cap. 1 par. 3.4.1.

<sup>6</sup>Cfr., BUCHANAN B., FEIGENBAUM E., *Dendral and Metadendral: their Application Dimension*, in "Artificial Intelligence", 2, pp. 5-24, 1978.

prevalentemente la conoscenza dichiarativa, la descrizione del compito da svolgere e del suo contesto. Ciò consente di usare il medesimo motore inferenziale per applicazioni diverse.

Tra i principali gusci, possiamo ricordare EMYCIN, creato estraendo dal sistema esperto MYCIN - specializzato nella diagnosi di malattie infettive - le parti non legate a quel compito particolare<sup>7</sup>.

Un ulteriore passo verso una maggiore generalità si ebbe con la proposta di *tool kits* (letteralmente, cassetta di attrezzi) che comprendono strumenti tratti da più gusci diversi.

I sistemi ispirati alla sintesi appena illustrata rappresentano un compromesso tra le esigenze alla base dei due approcci summenzionati. Quanto più tendono alla generalità, tanto più si ripresenta il problema dell'esplosione combinatoria, e si richiede una grande potenza di calcolo per ottenere un'efficienza accettabile. Quanto più sono specifici, tanto maggiore è lo sforzo intellettuale richiesto da ogni nuova applicazione, e minore la flessibilità del sistema.

#### 4. La programmazione logica (cenni)

In Europa, la fase menzionata nel paragrafo 3.2 precedente non ebbe l'importanza che rivestì negli Stati Uniti. Le ricerche di intelligenza artificiale continuarono a rivolgersi soprattutto verso l'elaborazione di metodi e tecniche ispirate alla logica.

Tali ricerche condussero allo sviluppo della programmazione logica in senso stretto, cioè all'uso di un sottoinsieme determinato della logica dei predicati, le clausole di Horn, come linguaggio di programmazione<sup>8</sup>. La più importante realizzazione in questa direzione fu un nuovo linguaggio di programmazione, il PROLOG. Il PROLOG, sul quale ci diffonderemo più ampiamente nel seguito, è un linguaggio di programmazione particolare: riprende taluni aspetti dalla logica dei predicati, altri dalla programmazione tradizionale: in questo senso è un linguaggio di programmazione logica.

Come i sistemi realizzati sulla base degli approcci menzionati sub 2 e 3, anche un programma in PROLOG

---

<sup>7</sup>Cfr. BUCHANAN B., SHORTLIFFE E., *Rule-Based Expert Systems*, Addison-Wesley, Reading (Massachusetts), 1984.

<sup>8</sup>Sulla programmazione logica e, in particolare, sul PROLOG, cfr. cap. 7 par. 2.

rappresenta una soluzione di compromesso tra la tendenza alla generalità e l'uso di metodi di ragionamento specifici: esso unisce aspetti dichiarativi e aspetti procedurali, sfrutta un metodo generale di inferenza incorporato nello stesso PROLOG, ma in una sequenza definita dal programma. La relazione con la logica matematica consente tuttavia - in contrasto con il carattere prevalentemente pragmatico di quegli approcci - di dare una solida base teorica alla programmazione (ad esempio, la semantica del programma, ciò che esso significa, può essere studiata a prescindere dal particolare sistema di calcolo, nei termini della teoria logico-matematica dei modelli). Il PROLOG, benché sviluppato per applicazioni nei settori del trattamento del linguaggio naturale e della dimostrazione automatica di teoremi, ha trovato applicazioni sempre più ampie<sup>9</sup>.

Si tratta di un linguaggio di tipo generale: come negli altri linguaggi di programmazione evoluti, in linea di principio, ogni algoritmo è esprimibile anche in PROLOG; numerosi problemi, in particolare per applicazioni di intelligenza artificiale, si possono tuttavia formulare in modo molto più semplice che nei linguaggi tradizionali.

Come si è già osservato, un programma PROLOG unisce aspetti dichiarativi e aspetti procedurali. Il programma può essere interpretato come:

(a) un insieme di enunciati che descrivono il problema da risolvere e il suo contesto (una base di conoscenza). Il sistema PROLOG (che opera come un motore inferenziale, in base ad uno schema logico generale deduce la soluzione del problema da questa descrizione.

(b) un insieme di procedure, la cui esecuzione realizza la soluzione del problema.

La considerazione dell'aspetto dichiarativo consente una chiara definizione del problema, la considerazione dell'aspetto procedurale consente di tener conto degli aspetti di efficienza.

Un programma in PROLOG, in un certo senso, è già un sistema di conoscenza. Un uso diretto del PROLOG presenta tuttavia taluni problemi:

-L'unione di aspetti dichiarativi e procedurali, può essere utile in un linguaggio di programmazione, ma può rivelarsi controproducente nello sviluppo di una base di conoscenza. Tenuto conto delle diverse competenze dei soggetti coinvolti (esperto, ingegnere della conoscenza, programmatore) nella realizzazione di un sistema esperto, è preferibile che l'esperto sia il più possibile autonomo nella creazione e nell'aggiornamento della base di conoscenza. A tal fine sarebbe utile consentirgli di

---

<sup>9</sup>In particolare, è stato scelto nel famoso progetto giapponese degli elaboratori della quinta generazione.

astrarre dagli aspetti procedurali, per concentrarsi sulla formalizzazione delle conoscenze rilevanti. Conseguentemente sarebbe interessante poter incorporare le conoscenze procedurali in un motore inferenziale che sia, più specifico, cioè adeguato al particolare tipo di problema, ma che consenta metodi di formalizzazione della conoscenza indipendenti dall'uso della conoscenza stessa<sup>10</sup>.

-Il linguaggio delle clausole di Horn, sul quale è basato il PROLOG, può sembrare insufficiente per la rappresentazione di conoscenze complesse.

Una soluzione può consistere nell'uso del PROLOG (o un altro linguaggio di programmazione logica) come un metalinguaggio, uso per il quale il PROLOG è particolarmente adatto (come si osservava, tra le prime applicazioni del PROLOG vi fu l'analisi del linguaggio naturale). Il PROLOG può cioè essere usato per la metaprogrammazione, cioè per sviluppare programmi in grado di manipolare conoscenza formulata in diversi linguaggi.

Possiamo facilmente scrivere in PROLOG un interprete<sup>11</sup> per il PROLOG o per altri linguaggi. Il motore inferenziale scritto in PROLOG consisterà di metaconoscenza (regole relative all'uso della conoscenza), che indica il modo nel quale enunciati del linguaggio di partenza (della base di conoscenza), debbono essere trasformati in enunciati direttamente eseguibili da parte del motore inferenziale del PROLOG (cioè in enunciati, o clausole, in PROLOG). La tecnica della metaprogrammazione è uno strumento estremamente flessibile, che consente di realizzare qualsiasi motore inferenziale, in molti casi con un'efficienza accettabile. Usando questa tecnica sono stati scritti, ad esempio, programmi che riproducono il funzionamento dei motori inferenziali di alcuni tra i più noti sistemi esperti, come, ad esempio, il già menzionato EMYCIN.

La programmazione logica è ancora agli inizi (il primo sistema PROLOG fu realizzato alla metà degli anni 70). Il PROLOG fu concepito dai suoi stessi inventori come una soluzione provvisoria, una prima approssimazione nella direzione dell'uso della logica come linguaggio di programmazione. Invece, esso ha trovato un enorme successo e si è rivelato adeguato per un'ampia gamma di applicazioni.

Tuttavia, non sono mancate le ricerche volte a modificare il meccanismo di deduzione del PROLOG o ad

---

<sup>10</sup>In realtà, risulta difficile prescindere completamente dall'aspetto procedurale.

<sup>11</sup>Un interprete è un programma che accetta come input gli enunciati di un certo linguaggio e li trasforma in enunciati - o "istruzioni", come si suole dire - direttamente eseguibili da parte del sistema di calcolo.

ampliare il linguaggio della programmazione logica<sup>12</sup>. Queste ricerche sono particolarmente importanti al fine dello sviluppo di applicazioni giuridiche, poiché il diritto è un settore particolarmente complesso, e rispetto ad esso alcuni limiti del PROLOG appaiono con maggiore evidenza<sup>13</sup>.

---

<sup>12</sup>Illustreremo ampiamente, nel cap. 8 par. 2.3, una recente proposta in questa direzione, sviluppata da L.T. McCarty tenendo conto proprio delle applicazioni informatico-giuridiche .

<sup>13</sup>Alcuni problemi della rappresentazione della conoscenza giuridica in PROLOG saranno considerati nel cap. 8 par. 6.



## CAP. 4

# I SISTEMI BASATI SULLA CONOSCENZA GIURIDICA

Negli ultimi anni si è assistito alla progettazione e alla realizzazione di numerosi sistemi informatico-giuridici che ricorrono ai metodi dell'intelligenza artificiale<sup>1</sup>.

Sono molto più numerosi i programmi qualificati come *sistemi esperti giuridici*. Bisogna osservare a questo proposito che:

(a) Nessuno di questi programmi unisce tutti gli elementi che compongono il concetto di sistema esperto<sup>2</sup>. Alcuni di essi presentano le caratteristiche strutturali dei sistemi esperti (si tratta di sistemi di conoscenza), nessuno presenta le caratteristiche funzionali dei sistemi esperti (cioè l'attitudine a fornire prestazioni che richiedano notevoli competenze).

(b) Nessuna applicazione giuridica dell'intelligenza artificiale si è rilevata suscettibile di una concreta utilità nel lavoro giuridico, così da consentire uno sfruttamento commerciale significativo<sup>3</sup>.

<sup>1</sup>Tra le rassegne più recenti ed aggiornate delle applicazioni giuridiche dell'intelligenza artificiale, cfr. SUSSKIND R.E., *Expert Systems in Law*, cit., specialmente pp. 257-261; BIAGIOLI C., FAMELI E., *\*Expert Systems in Law: An International Survey and a Selected Bibliography*, Istituto per la documentazione giuridica, Firenze, 1987 (in pubblicazione in "CCAI (the Journal for the Integrated Study of Artificial Intelligence, Cognitive Science and Applied Epistemology)", 4, 3; MARTINO A.A., *I sistemi esperti legali*, relazione presentata al congresso "Il processo civile: oltre l'automazione verso i sistemi esperti" (Roma, maggio 1987); SERGOT M., *The Representation of Law in Computer Programs*, cit. Nella redazione di questo capitolo ho tenuto conto soprattutto di quest'ultimo lavoro.

<sup>2</sup>Secondo la definizione da noi accolta (cfr. cap. 2 par. 2).

<sup>3</sup>Bisogna lamentare l'uso non sempre onesto del linguaggio dell'intelligenza artificiale. Come già si accennava, spesso si ritiene di poter nobilitare un programma informatico tradizionale o un archivio di dati qualificandoli rispettivamente come *expert system* e come *knowledge base*, in modo da sfruttare la connotazione di modernità, di avanguardia tecnologica, che molti associano all'intelligenza artificiale. Questo discutibile uso linguistico sembra particolarmente diffuso nell'informatica giuridica, dove rappresenta non solo una forma di pubblicità (menzognera) per prodotti commerciali, ma anche l'abitudine di taluni studiosi della materia.

Si potrebbe forse ammettere l'uso del termine *sistema esperto giuridico* (usato senza connotazioni valutative), anche per sistemi che possiedano solo le caratteristiche strutturali dei sistemi esperti. Tuttavia, seppure in contrasto con l'uso linguistico, sembra preferibile evitare questo termine (e non per indicare lontane ed improbabili prospettive). Questa espressione induce a ritenere che il sistema informatico così chiamato sia un sostituto del giurista esperto, mentre, come vedremo, non solo un sistema esperto giuridico non è ancora stato realizzato, ma sistemi di questo tipo non sembrano possibili allo stato della tecnica.

Si userà invece l'espressione *sistemi basati sulla conoscenza giuridica* per indicare i sistemi di conoscenza che includano una rappresentazione di contenuti giuridici (distinta dai programmi che usano questi contenuti), e possano quindi essere usati per trattare problemi giuridici.

Riprendendo l'analisi sviluppata da L.T. McCarty<sup>4</sup>, possiamo distinguere tre categorie fondamentali di sistemi informatico-giuridici "intelligenti": i sistemi per l'analisi giuridica, i sistemi per la pianificazione giuridica e i sistemi per il reperimento di informazioni concettuale:

(a) I sistemi per l'analisi giuridica rispondono a domande sulla qualificazione giuridica di una determinata situazione. Tipicamente, un sistema di questo tipo "accetta come input la descrizione di un caso, e fornisce come output una classificazione del caso nei termini di varie categorie giuridiche, ad esempio, il sistema conclude che il cliente deve pagare una tassa sull'incremento di capitale"... "Esso individua le regole giuridiche rilevanti e suggerisce un'analisi di un insieme di fatti determinato"<sup>5</sup>

(b) I sistemi per la pianificazione giuridica indicano come può essere ottenuto un certo risultato giuridico, a partire da una data situazione, soddisfacendo determinate esigenze. Nei sistemi di questo tipo "l'insieme dei fatti rilevanti è variabile": il giurista descrive una situazione iniziale, ad esempio, una società per azioni esistente, e un risultato finale desiderato, ad esempio, l'acquisto di certi beni, e il sistema esamina lo spazio dei possibili tipi di transazione per suggerire un corso d'azione che soddisfi

---

<sup>4</sup>MCCARTHY T., *Intelligent Legal Information Systems: Problems and Prospects*, in "Routgers Computer & Technology- Law Journal", 9, 1983, pp. 265-294; MCCARTY T., *Intelligent Legal Information Systems: an Update*, in "Law and Computers", 5, 1987, pp. 196-202 (Law and Computers Association of Japan).

<sup>5</sup>MCCARTY L.T., *Intelligent Legal Information Systems: an Update*, cit., p. 196.



certi requisiti aggiuntive, ad esempio, le minime conseguenze fiscali<sup>6</sup>.

(c) I sistemi per il *reperimento concettuale di informazioni giuridiche* ricercano dette informazioni in base ad una considerazione della loro rappresentazione formale in strutture concettuali. In questi sistemi si utilizza "il modello concettuale di un particolare settore del diritto per costruire la base di dati che contiene le informazioni rilevanti rispetto ad un certo insieme di casi"<sup>7</sup>. La ricerca delle informazioni, anziché in base al confronto di due gruppi di vocaboli (i vocaboli usati nella domanda e quelli contenuti nei testi memorizzati nella base di dati), avviene in base al confronto di due insiemi di strutture concettuali (le strutture che corrispondono ai contenuti concettuali espressi nella domanda e le strutture che corrispondono ai contenuti dei testi giuridici memorizzati nella base di dati).

I sistemi informatico-giuridici intelligenti, sotto il profilo funzionale, possono avvicinarsi ad alcuni dei programmi che abbiamo menzionato nel cap. 2. In particolare, i sistemi per il reperimento di informazioni concettuale rappresentano lo sviluppo delle banche di dati giuridici, mentre i sistemi per la pianificazione e l'analisi giuridica rappresentano lo sviluppo dei sistemi per l'applicazione automatica di norme giuridiche, e, in particolare, dei "sistemi per consulenze giuridiche" che abbiamo presentato nel cap. 2 par. 2.1.2. I sistemi "intelligenti" si differenziano da questi ultimi, anche quando le prestazioni siano analoghe, in quanto usano metodi dell'intelligenza artificiale e, in particolare, presentano la struttura dei sistemi basati sulla conoscenza.

In questa sede ci limiteremo alla breve presentazione di alcuni dei progetti più significativi, senza pretesa di completezza, ma al solo fine di illustrare sinteticamente realizzazioni e prospettive dell'intelligenza artificiale nel diritto.

Saranno usati alcuni concetti dell'intelligenza artificiale. Chi non abbia alcuna familiarità con questa tematica può far precedere la lettura dei cap. 5, 6 e 7 seguenti, a quella del presente capitolo.

---

<sup>6</sup>MCCARTY L.T., *Intelligent Legal Information Systems: an Update*, cit., p. 198.

<sup>7</sup>Cfr. MCCARTY L.T., *Intelligent Legal Information Systems: Problems and Prospects*, cit., p. 276. Per modello concettuale si intende una rappresentazione formale della conoscenza utilizzabile da un sistema automatico. Il problema della rappresentazione della conoscenza nell'intelligenza artificiale sarà trattato ampiamente nella seconda parte del presente volume.

Nella presentazione dei diversi progetti si seguirà prevalentemente l'ordine cronologico, ma si presenteranno unitariamente progetti sviluppatisi nell'arco di più anni, anche quando ciò comporti una sovrapposizione con progetti successivi.

## 1. I predecessori (cenni)

Già in alcuni precursori dell'informatica giuridica possiamo trovare interessanti riferimenti a sviluppi che solo oggi, grazie all'intelligenza artificiale, stanno diventando attuali. Inoltre, possiamo trovare riferimenti a prospettive e problematiche dei sistemi informatico-giuridici intelligenti nell'opera di numerosi filosofi e teorici del diritto, sia in autori di ispirazione razionalistica, che da sempre cercato di ricondurre il diritto ad una dimensione assiomatica (e talvolta anche formale), sia in quelli che, opponendosi a questo approccio, hanno contribuito ad precisare condizioni e limiti delle applicazioni giuridiche dell'informatica e, in particolare, dell'intelligenza artificiale<sup>8</sup>.

La prima discussione di problematiche informatico-giuridiche che movesse consapevolmente dalle prospettive dell'intelligenza artificiale è forse rappresentata<sup>9</sup> da un articolo di Buchanan e Headrick, pubblicato nel 1970, dal titolo "Alcune riflessioni su intelligenza artificiale e ragionamento giuridico"<sup>10</sup>. Gli autori partivano dall'esame dell'attività dell'operatore giuridico (in particolare, l'avvocato) e individuavano due tipi di situazioni nelle quali il giurista è chiamato a prestare la propria opera:

-Dopo che i fatti si sono verificati, il giurista deve informare il cliente sulla sua posizione giuridica, e predisporre l'argomentazione più favorevole a quest'ultimo. Buchanan e Headrick ponevano l'accento sul problema della costruzione dell'argomentazione, e osservavano che, rispetto a questo problema, raramente vi sono soluzioni univoche.

-Prima che i fatti si siano verificati, il giurista può essere chiamato a consigliare al cliente il tipo d'azione

---

<sup>8</sup>Il problema del rapporto tra intelligenza artificiale e tradizione filosofico-giuridica, sarà affrontato più ampiamente nel cap. 6.

<sup>9</sup>Come afferma GARDNER A., *Law Applications*, in SHAPIRO S.C., ECKROTH D., *Encyclopedia of Artificial Intelligence*, cit., 1987, pp. 457.

<sup>10</sup>BUCHANAN B.G., HEADRICK T.E., *Some Speculations about Artificial Intelligence and Legal Reasoning*, in "Stanford Law Review", 23, 1970, pp. 40-62.

più opportuno per raggiungere i suoi scopi (giuridici, economici ecc.). Si tratta di predisporre il piano d'azione più adeguato sotto il profilo della massimizzazione dell'utilità del cliente (della valutazione comparativa dell'idoneità a raggiungere quegli scopi e della possibilità di pregiudicare altre esigenze del cliente stesso)<sup>11</sup>.

Per valutare la possibilità di realizzare applicazioni giuridiche dell'intelligenza artificiale Buchanan e Headrick esaminavano i principali sistemi esperti sviluppati fino ad allora e cercavano di porre in rapporto i metodi del ragionamento giuridico e le tecniche dell'intelligenza artificiale rispetto a problemi come la pianificazione di scopi, il riconoscimento di fatti rilevanti, la soluzione delle antinomie, l'uso dell'analogia. Secondo questi autori la realizzazione di applicazioni giuridiche dell'intelligenza artificiale dipende dalla possibilità di ridurre i problemi giuridici ad un preciso contesto strutturato, riduzione che era stata alla base del relativo successo di alcuni sistemi esperti<sup>12</sup>.

Il lavoro di Buchanan e Headrick ha il pregio di indicare con chiarezza da un lato le ambiziose prospettive, dall'altro i limiti dell'intelligenza artificiale nel diritto. I problemi giuridici sono complessi, non deterministici, scarsamente strutturati, il loro contesto è il mondo della vita e dell'esperienza<sup>13</sup> piuttosto che ambiti predefiniti e formalizzati. Un vero sistema esperto

---

<sup>11</sup>Si tratta, sostanzialmente, della distinzione tra sistemi di consultazioni giuridica e di sistemi di analisi giuridica che abbiamo illustrato sopra.

<sup>12</sup>Cfr. GARDNER A., *Law Applications*, cit., p. 457. Sul rapporto tra strutturazione o programmabilità dei problemi e uso dell'informatica, cfr. SIMON H.A., *Informatica, direzione aziendale e organizzazione del lavoro*, Franco Angeli, Milano, 1979 (traduzione di *The New Science of Management Decisions*, Prentice-Hall, Englewood Cliffs (New Jersey), 1977), p. 52 ss.

<sup>13</sup>Ricordiamo il significato di questi concetti nelle tradizioni della fenomenologia e dell'esistenzialismo, alle quali l'intelligenza artificiale sta guardando con crescente interesse. Cfr., per una considerazione del rapporto tra queste correnti filosofiche e l'intelligenza artificiale, MALLERY J.C., HURWITZ R., *Hermeneutics*, in SHAPIRO S.C., ECKROTH D., *Encyclopedia of Artificial Intelligence*, cit.; HALL H., *Phenomenology*, in SHAPIRO S.C., ECKROTH D., *Encyclopedia of Artificial Intelligence*, cit., pp. 730-736. Si veda anche WINOGRAD T., FLORES F., *Understanding Computer and Cognition*, cit., un lavoro pieno di spunti interessanti, anche se in un quadro concettuale complesso, nel quale esistenzialismo ed ermeneutica si fondono con le concezioni "autopoietiche" di Maturana e Varela.

giuridico (un sistema informatico in grado di esibire una competenza analoga a quella di un esperto giurista) differirebbe radicalmente da tutte le applicazioni informatiche finora sviluppate. Esso rappresenterebbe una rivoluzione scientifica e la dimostrazione della possibilità di una estensione tendenzialmente illimitata dell'informatizzazione.

Un sistema siffatto non sembra realizzabile (almeno nell'ambito delle tecnologie oggi disponibili e dei loro prossimi sviluppi). Pertanto chi intenda sviluppare utili applicazioni dell'intelligenza artificiale a problemi giuridici deve rinunciare al progetto di riprodurre il giurista in un sistema automatico, e cercare piuttosto di isolare settori e aspetti determinati dell'attività giuridica nei quali sia possibile il ricorso ai metodi dell'intelligenza artificiale.

## 2. I principali progetti degli anni 70

Nel presente paragrafo esamineremo i progetti pionieristici, avviati negli anni 70. Come abbiamo già osservato, ci limiteremo ai sistemi che usano tecniche di intelligenza artificiale.

### 2.1. Il progetto TAXMAN

Il progetto TAXMAN, forse la più celebre tra le applicazioni giuridiche dell'intelligenza artificiale, fu avviato da L.T. McCarty nel 1972<sup>14</sup> e, con alcune interruzioni, è continuato fino ad oggi. Le due versioni del sistema (TAXMAN I e TAXMAN II) si sono occupate dello stesso settore del diritto, la disciplina fiscale delle società per azioni.

TAXMAN I, fu sviluppato nella prima metà degli anni 70, usando il linguaggio Micro-PLANNER<sup>15</sup>. Il sistema aveva per oggetto la tassazione delle riorganizzazioni delle società per azioni, e consentiva di determinare, in base alla descrizione dei fatti rilevanti, se si fosse verificata una

---

<sup>14</sup>Anche se la prima versione di TAXMAN (TAXMAN I) fu realizzata all'inizio degli anni 70, essa fu descritta solo in MCCARTY T., *Reflections on TAXMAN: An Experiment in Artificial Intelligence and Legal Reasoning*, in "Harvard Law Review", 90, 1977, pp. 837-893.

<sup>15</sup>Micro-PLANNER, fu implementato da Sussman, Winograd e Charniak all'inizio degli anni 70. Questo sistema unisce al LISP un dimostratore di teoremi, anticipando alcuni aspetti del PROLOG.

trasformazione di società esente da imposta ai sensi dell'Internal Revenue Code<sup>16</sup>.

TAXMAN I rappresentava la conoscenza mediante reti semantiche (al livello della logica predicativa), e eseguiva inferenze deduttive<sup>17</sup>.

I limiti di TAXMAN I e, in particolare, la difficoltà di rappresentare contesti complessi (soprattutto gli aspetti temporali) e di trattare concetti indeterminati (*open textured* o *amorphous* nella terminologia di McCarty), inducevano McCarty<sup>18</sup> a sviluppare un nuovo sistema, TAXMAN II, nella seconda metà degli anni 70. TAXMAN II usa il linguaggio AIDMS, basato su *frames*<sup>19</sup>, ed è stato applicato a diversi aspetti della legislazione fiscale delle società per azioni, come il concetto di reddito in relazione ad un emendamento della costituzione americana che nel 1913 autorizzava un'imposta federale sui redditi<sup>20</sup>.

Per trattare i concetti indeterminati<sup>21</sup>, TAXMAN II ricorre ai concetti di *prototipo* e *deformazioni* (*prototype and deformations*)<sup>22</sup>. I concetti giuridici (ad esempio, il concetto di diritto azionario) anziché da condizioni necessarie e sufficienti del loro impiego possono essere caratterizzati nei termini di:

-un insieme di paradigmi o prototipi (i prototipi di azione ordinaria, di azione privilegiata, di obbligazione,

<sup>16</sup>Gli artt. 354-365, 358, 361-362 dell'Internal Revenue Code disciplinano i casi nei quali le riorganizzazioni di società sono esenti da tassazione (ad esempio, nel caso di trasferimento di azioni da una società ad altra società che viene a controllare la prima società immediatamente in seguito al trasferimento).

<sup>17</sup>La rappresentazione della conoscenza di TAXMAN I è illustrata nel cap. 8 par. 5.

<sup>18</sup>MCCARTY T., *The TAXMAN Project: Towards a Cognitive Theory of Legal Argument*, in NIBLETT B. (a cura di), *Computer Science and Law: An Advanced Course*, cit. 1980, pp. 23-43

<sup>19</sup>SHIDARAN N.S., *Representing Knowledge in AIDMS*, in "Informatica e diritto", 7, 1981, pp. 201-221.

<sup>20</sup>Cfr. MCCARTY L.T., *A Computational Theory of Eisner v. Macomber*, cit.

<sup>21</sup>Nel settore scelto da McCarty si tratta di concetti come quelli di *continuità di interesse*, *scopo dell'impresa*, *reddito* ecc.

<sup>22</sup>MCCARTY T., SHIDARAN N.S., *The Representation of Evolving Systems of Legal Concepts*, Technical Report LPR-TR-13, Computer Science Department, Rutgers University, 1981. Il concetto di prototipo sarà approfondito nel cap. 5.

ciascuno caratterizzato da un insieme determinato di diritti rispetto agli utili, al patrimonio sociale, al controllo) e

- un insieme di trasformazioni che consentono di passare dai prototipi ai casi concreti (questi possono presentare solo alcuni degli elementi compresi nell'uno o nell'altro prototipo) e da un prototipo all'altro<sup>23</sup>.

In questo modo diverrebbe possibile considerare non solo quando un caso concreto sia un'istanza di un prototipo (ad esempio, sia un'azione ordinaria, un'azione privilegiata o un'obbligazione), ma anche se, e in quale misura, il caso si avvicini ad un diverso prototipo e debba essere assoggettato alla disciplina prevista per quest'ultimo (ad esempio, quando il trattamento di diritti obbligazionari possa essere assimilato a quello dei diritti azionari).

Poiché un caso determinato può essere rapportato a diversi prototipi, a ciascuno dei quali può corrispondere una diversa disciplina giuridica<sup>24</sup>, il modello proposto da McCarty si presta ad essere usato per generare diverse argomentazioni, che conducano a soluzioni alternative del caso.

Ad esempio, McCarty usa il modello di prototipo e deformazione per generare le diverse argomentazioni sviluppate dalla maggioranza e dalle *dissenting opinions* rispetto ad un'importante sentenza in materia di imposte sul reddito, la causa *Eisner v. Macomber*, del 1920. In questa causa, la Corte Suprema degli Stati Uniti si trovò a decidere se una distribuzione di azioni ai soci rappresentasse reddito (*income*) - come nel caso prototipico della distribuzione di dividendi in denaro - e potesse quindi essere tassata ai sensi del sedicesimo emendamento della costituzione americana<sup>25</sup>, o se rappresentasse invece un aumento di valore di un bene - come nel caso prototipico di un mero incremento del valore di mercato delle azioni - e

---

<sup>23</sup>Più esattamente, un concetto *amorphous* può essere definito da tre elementi (a) un eventuale invariante, cioè l'insieme delle condizioni necessarie del concetto; (b) un insieme di prototipi del concetto; (c) un insieme di trasformazioni che consentono di passare da un prototipo all'altro (le trasformazioni rappresentano l'elemento unificante del concetto).

<sup>24</sup>Ad esempio, con una trasformazione incrementale delle posizioni giuridiche (diminuendo le posizioni che implicano rischi e i poteri di controllo) sarebbe possibile trasformare il prototipo del possesso di azioni in quello del possesso di obbligazioni (cfr. MCCARTY T., *The TAXMAN Project: Towards a Cognitive Theory of Legal Argument*, cit., p. 31 ss.).

<sup>25</sup>Il sedicesimo emendamento consentiva l'imposizione federale solo sui redditi (non sulla proprietà).

la sua tassazione costituisse un'inammissibile imposta sulla proprietà<sup>26</sup>.

La ricerca di McCarty è stata sempre ispirata dall'esigenza di sviluppare sistemi basati su un'approfondita analisi del pensiero giuridico, su un modello concettuale profondo del diritto<sup>27</sup>. Nel progetto TAXMAN, per raggiungere questo obiettivo McCarty aveva fatto ricorso ai linguaggi per la rappresentazione della conoscenza usati dall'intelligenza artificiale negli anni 70. Questi linguaggi si sono però rilevati insufficienti rispetto all'ambizioso progetto di McCarty, viziati da una mancanza di rigore logico. Il relativo insuccesso di TAXMAN è alla base di un nuovo orientamento di McCarty, che ha recentemente sviluppato un linguaggio per il discorso giuridico (*Language for Legal Discourse-LLD*) ispirato alla programmazione logica (anche se riprende alcuni aspetti di TAXMAN II), come quadro unificante per lo sviluppo di sistemi basati sulla conoscenza giuridica<sup>28</sup>.

## 2.2. Il progetto CLLIPS

All'inizio degli anni 70, P.B. Maggs e G.C. Debessonnet, avviarono una ricerca volta a sviluppare un linguaggio formale per la rappresentazione degli ordinamenti giuridici<sup>29</sup>. Nell'ambito di questa ricerca gli autori proposero un programma (in LISP) che operava inferenze della logica proposizionale a partire da una rappresentazione normalizzata di un testo giuridico.

Debessonnet proseguì le ricerche nel campo della rappresentazione formale di testi giuridici, avviando alla fine degli anni 70, con la collaborazione di Cross, il progetto CCLIPS (*Civil Code Legal Information Processing System*). CCLIPS ha per oggetto il codice civile della Louisiana, e in particolare, la disciplina dei contratti<sup>30</sup>.

---

<sup>26</sup>Cfr. MCCARTY T., 1982, *A Computational Theory of Eisner v. Macomber*, cit.

<sup>27</sup>Il concetto di modello concettuale profondo sarà esaminato nel cap. 6 par. 7.

<sup>28</sup>Il linguaggio per la rappresentazione della conoscenza proposto da McCarty sarà illustrato nel cap.8 par. 2.3. -

<sup>29</sup>MAGGS P.B., DE BESSONET C.G., *Automated Logical Analysis of Systems of Legal Rules*, in "Jurimetrics Journal", 12, 1972, pp. 150-169.

<sup>30</sup>Cfr. CROSS G.R., DE BESSONET G.R., *Legislative Techniques and Automated Systems*, in Atti del terzo congresso su L'informatica

E' stato sviluppato un linguaggio per la rappresentazione della conoscenza giuridica basato fondamentalmente sulla logica proposizionale, la forma normalizzata atomica (ANF-Atomically Normalized Form). Gli enunciati sono espressi nei termini di elementi primitivi (che esprimono concetti corrispondenti ad oggetti ed azioni) volti a rappresentare la struttura semantica profonda del testo analizzato. Usando questo linguaggio Cross e De Bessonnet intendono realizzare un sistema che consenta di effettuare forme di reperimento di informazioni concettuale e inferenze della logica proposizionale. Il progetto CCLIPS si propone inoltre predisporre strumenti per un miglioramento della tecnica legislativa, per una "legislazione scientifica", come affermano gli stessi autori.

### 2.3. Il sistema JUDITH

Nella prima metà degli anni 70 fu sviluppato in Europa il sistema JUDITH, che fu applicato ad alcune norme del codice civile tedesco in materia di responsabilità civile<sup>31</sup>. JUDITH era implementato in FORTRAN e, riprendendo alcuni aspetti di MYCIN, si basava su regole di produzione espresse nella logica proposizionale. Un'interfaccia interattiva consentiva di un dialogo tra sistema e l'utente. Come in molti sistemi sviluppati in seguito, il sistema interrogava l'utente sulle condizioni cui era subordinata una data situazione giuridica. L'utente poteva rispondere a ogni domanda con *si*, *no*, *non so*. Se rispondeva *non so* il sistema considerava se la condizione potesse considerarsi verificata in base a regole comprese nella base di conoscenza.

### 2.4. Il progetto di Meldman

Un ambizioso progetto di sistema basato sulla conoscenza giuridica fu sviluppato da Meldman in una tesi di dottorato presentata al MIT nel 1975<sup>32</sup>.

---

giuridica e la comunità nazionale ed internazionale, Roma, 1983; CROSS G.R., DE BESSONNET C., *Representation of Some Aspects of Legal Causality*, in WALTER C. (a cura di), *Computing Power and Legal Reasoning*, cit., pp. 205-214; CROSS G.R., DE BESSONNET C.G., *Conceptual Retrieval and Legal Decision Making*, in MARTINO A.A., SOCCI NATALI F. (a cura di) *Automated Analysis of Legal Texts*, cit., pp. 219-227.

<sup>31</sup>POPP W.G., SCHLINK B., *JUDITH: a Computer Programme to Advise Lawyers in Reasoning a Case*, in "Jurimetrics Journal", 15, 1975, pp. 303-314.

<sup>32</sup>MELDMAN J.A., *A Preliminary Study in Computer-aided Analysis*, MIT Report MAC TR-157, M.I.T., Cambridge (Mass) Novembre 1975; MELDMAN J.A.,



Il settore del diritto esaminato da Meldman era la disciplina della responsabilità per aggressioni e percosse (*assault and battery*), settore che negli Stati Uniti d'America è regolato dal *case law*, cioè dalla disciplina sviluppata mediante pronunce giudiziali, secondo i principi dei sistemi di *common law*.

Come è noto, nei sistemi di *common law*, in mancanza di una norma legislativa, le nuove decisioni debbano essere fondate con riferimento ai principi - *rationes decidendi* - già usati in casi precedenti. Pertanto Meldman distinse tra due tipi di conoscenze giuridiche da rappresentare nel sistema:

(a) norme estratte da singoli precedenti giudiziari (la *ratio decidendi* della decisione del precedente) e

(b) norme, in genere più astratte, formulate nella ricostruzione dottrinale della materia (estratte da un libro di testo di Prosser in materia di responsabilità civile).

Uno degli aspetti più interessanti del sistema è un trattamento dell'analogia. La rappresentazione della conoscenza giuridica includeva una classificazione concettuale di tipo gerarchico<sup>33</sup>. Le *rationes decidendi* dei precedenti potevano trovare applicazione non solo quando il nuovo caso corrispondeva esattamente alla regola estratta dal precedente, ma anche quando il caso fosse descritto con un predicato più specifico di quello usato nella regola e soprattutto (qui il ragionamento diventa analogico), quando sia il caso che la regola fossero specificazioni dello stesso concetto generale.

Si osservino, tuttavia, i limiti del trattamento del ragionamento analogico nel progetto di Meldman, limiti che nessun programma sviluppato in seguito ha saputo realmente superare<sup>34</sup>:

-Il sistema considera analoghi tutti i casi che presentino concetti dello stesso genere (compravendita e mediazione sono contratti e pertanto la disciplina della mediazione potrebbe essere estesa alla compravendita). Tuttavia, esso non considera le speciali ragioni che, in un caso particolare, possono far ritenere opportuno andare al di là della formulazione letterale di una norma per estenderne la disciplina ad altri casi. Vengono pertanto

---

A *Structural Model for Computer Aided Legal Analysis*, in "Rutgers Journal of Computers and the Law", 1977, pp. 27-71.

<sup>33</sup>Si trattava di una gerarchia concettuale del tipo *è un*, che illustreremo nel cap. 5 par. 2.

<sup>34</sup>Un apparato concettuale più articolato per il trattamento dell'analogia è offerto dal paradigma di prototipo e trasformazione di McCarty.

ritenute analoghe numerose situazioni che nessun giurista riterrebbe tali.

-La somiglianza tra due casi dipende dalla rappresentazione formale del diritto che costituisce la base di conoscenza del sistema. Tale rappresentazione formale che predetermina sussunzioni e analogie. Sia la formalizzazione delle norme giuridiche (delle *rationes decidendi* e delle regole "dottrinali"), sia la costruzione della rete concettuale, sono effettuate in astratto, una volta per tutte, e non in relazione al caso particolare. In questo senso, anche il sistema di Meldman si basa su una rappresentazione del diritto rigidamente predeterminata. Pertanto, molti degli sviluppi il ragionamento analogico consente al giurista (quelli che postulano una reinterpreteazione dei testi rilevanti o una ristrutturazione concettuale) sono impossibili.

Il sistema di Meldman era stato progettato per essere implementato in un sistema chiamato OWL, che non fu mai completato. Un programma basato sull'approccio di Meldman fu invece realizzato da King nel 1976<sup>35</sup>.

## 2.5. Il progetto LEGOL

Il progetto LEGOL è stato sviluppato in Inghilterra, presso la London School of Economics, a partire dalla metà degli anni 70. Lo scopo teorico fondamentale del progetto era sviluppare tecniche formali per analizzare il comportamento amministrativo, le strutture delle organizzazioni e i sistemi informativi. Questo obiettivo condusse il gruppo di ricercatori coordinati da Ronald Stamper ad occuparsi di norme (come fattori che determinano la struttura e le funzioni delle organizzazioni) e quindi di diritto<sup>36</sup>.

Il progetto LEGOL, ha condotto alla realizzazione dei linguaggi NORMA (per la rappresentazione della conoscenza) e LEGOL (per la manipolazione della conoscenza), quali elementi di un sistema per la rappresentazione e l'uso di norme sociali (e, in particolare, giuridiche). Il linguaggio LEGOL si basa fondamentalmente sull'algebra relazionale<sup>37</sup>,

---

<sup>35</sup>KING J.J., *Analysis of a KRL Implementation of a Current Legal Reasoning Program Design* (non pubblicato), 1976.

<sup>36</sup>STAMPER R.K., *The Automation of Legal Reasoning: Problems and Prospects*, LEGOL Project Report L39, LSE Systems Research Group, London School of Economics and Political Science, London, 1976

<sup>37</sup>Cfr. JONES R., MASON P., STAMPER R., *Legol 2.0: A Relational Specification Language for Complex Rules*, in "Information Systems", 4,

che integra con strutture di controllo della programmazione strutturata<sup>38</sup> e con una rappresentazione di concetti temporali e deontici<sup>39</sup>.

Nell'ambito del progetto LEGOL fu elaborata una metodologia per la rappresentazione formale dei contesti amministrativi ispirata ad un "modello epistemologico semantico" che trae ispirazione dalle metodologie per la progettazione di basi di dati (oltre che da talune considerazioni di ispirazione comportamentista delle quali è difficile valutare il cui reale influsso). I contenuti giuridici vanno rappresentati nei termini di entità, concetti e loro relazioni, da individuarsi secondo metodi determinati.

Applicazioni dei metodi e dei linguaggi elaborati nel progetto LEGOL sono state sviluppate in diversi settori del diritto, come la legislazione sociale, la disciplina successoria, la normativa fiscale.

Negli ultimi anni il progetto LEGOL sembra aver incontrato alcune difficoltà di base:

(a) Come ha osservato Sergot, l'algebra relazionale (la tecnica per la rappresentazione e la manipolazione della conoscenza adottata in LEGOL) può essere tradotta facilmente nella programmazione logica, con il vantaggio di integrarla in un linguaggio più ricco e completo (non è più necessario ricorrere alle strutture di controllo della programmazione tradizionale). Pertanto, il linguaggio sviluppato nel quadro del progetto LEGOL (algebra relazionale + costrutti della programmazione strutturata) sembra superato anche per quanto attiene all'attitudine a rappresentare contesti giuridici<sup>40</sup>,

---

1979, pp. 293-305; STAMPER R.K., TAGG C., MASON P., COOK S., MARKS J., *Developing the LEGOL Grammar*, in CIAMPI (a cura di) *Artificial Intelligence and Legal Information Systems*, cit., pp. 357-287. L'algebra relazionale è un linguaggio che consente di manipolare le tabelle di dati (relazioni) che compongono una base di dati relazionale. Tale linguaggio consente di specificare i dati richiesti come il risultato di una serie di operazioni sulle tabelle preesistenti (estrazione di righe o di colonne, unione di righe di tabelle diverse ecc.).

<sup>38</sup>Cioè con strutture del tipo `if .. then ..` o del tipo `while .. do ..`.

<sup>39</sup>Cfr. STAMPER R.K., *LEGOL: Modelling Legal Rules by Computer*, in NIBLETT (a cura di), *Computer Science and Law*, cit., pp. 45-71; STAMPER R.K., *A Non-classical Logic Based on the Structures of Behaviour*, in MARTINO A.A., SOCCI NATALI F., *Automated Analysis of Legal Texts*, cit., pp. 115-139.

<sup>40</sup>Come è noto, la programmazione logica può essere considerata come un'estensione del modello delle basi di dati relazionali: le relazioni corrispondono ai fatti di un programma logico. I linguaggi di programmazione logica, come il PROLOG, consentono però di esprimere non

(b) E' dubbia l'opportunità di definire in astratto un unico metodo per la ricostruzione delle strutture concettuali della conoscenza giuridica (almeno allo stato della tecnologia, sembra preferibile un approccio più pragmatico<sup>41</sup>) e, soprattutto, è discutibile l'opportunità di ispirarsi alle metodologie per lo sviluppo del software e delle basi di dati, piuttosto che alla millenaria tradizione della cultura giuridica.

Nei suoi lavori più recenti, Stamper ha rivendicato la validità dell'approccio sviluppato in LEGOL, sviluppando un'acuta critica del paradigma "logicistico" oggi dominante nell'intelligenza artificiale<sup>42</sup>. Tuttavia, non è chiaro come le soluzioni informatiche sviluppate nell'ambito di LEGOL, alle quali Stamper continua a far riferimento, possano rappresentare un'alternativa rispetto a quel paradigma.

## 2.6. Logica e diritto nell'opera di L.A. Allen

Tra i pionieri delle applicazioni giuridiche dell'intelligenza artificiale dobbiamo ricordare L.E. Allen benché abbia realizzato applicazioni di intelligenza artificiale solo negli ultimi anni.

Allen ha il merito di aver promosso nell'ambiente anglosassone l'uso della logica come tecnica per analizzare e rappresentare contenuti giuridici<sup>43</sup>. In particolare, egli ha elaborato un interessante metodo per la rappresentazione

---

solo fatti ma anche regole. Anche le operazioni dell'algebra relazionale possono essere facilmente espresse nella programmazione logica. Per un sintetico esame del problema, e per alcune indicazioni bibliografiche, cfr., ad esempio, STERLING L, SHAPIRO E., *The Art of Prolog*, MIT Press, Cambridge (Mass.), 1987, p. 30 ss.

<sup>41</sup>SERGOT M.J., *Programming Law: LEGOL as a Logic Programming Language*, Department of Computing and Control, London, 1980

<sup>42</sup>STAMPER R.K., BACKHAUS J., ALTHAUSE K., *Expert Systems: Layers Beware*, relazione presentata al 4° Congresso internazionale sul tema "Informatica e regolamentazioni giuridiche", Roma, Maggio 1988; STAMPER R.K., *The Role of Semantics in Legal Expert Systems and Legal Reasoning*, relazione presentata al Congresso internazionale *Expert Systems in Law* (Bologna, maggio 1989). La critica di Stamper alla programmazione logica sarà considerata approfonditamente nel cap. 9 par. 2.2.

<sup>43</sup>ALLEN L.E., *Symbolic Logic. A Razor-Edged Tool for Drafting and Interpreting Legal Documents*, in "Yale Law Journal", 66, 1957, pp. 833-879.

di testi giuridici, la normalizzazione<sup>44</sup>. La normalizzazione di un testo giuridico consiste nel processo che, a partire da un testo in linguaggio naturale, conduce alla sua versione normalizzata, cioè ad una sua riformulazione più semplice e rigorosa, la versione normalizzata. Nella versione normalizzata si utilizzano espressioni sintattiche predefinite (che corrispondono a connettivi della logica proposizionale e ad operatori deontici).

Recentemente Allen e Saxon hanno proposto un complesso sistema logico per la rappresentazione formale di contesti giuridici<sup>45</sup>, che comprende una logica predicativa (ispirata alle logiche rilevanti)<sup>46</sup>, una logica modale e una logica deontica (che formalizza gli otto concetti giuridici hohfeldiani)<sup>47</sup>.

Gli stessi autori hanno sviluppato, nel 1984, un programma chiamato NORMALIZER, destinato a fungere da supporto al processo della normalizzazione. Il programma, a partire da formule logiche e enunciati atomici, redige automaticamente testi normalizzati in diverse versioni logicamente equivalenti<sup>48</sup>.

Nel 1988 Allen e Saxon hanno presentato invece un AUTOPROLOG, un sistema che consente di generare automaticamente, a partire da un testo normalizzato, un

---

<sup>44</sup>Tra i numerosi scritti di Allen sull'argomento, cfr. ALLEN L.E., *Una guida per redattori giuridici di testi normalizzati*, in "Informatica e diritto", 2, 1978, pp. 61-114; ALLEN L.E., *Language, Law and Logic: Plain Legal Drafting for the Electronic Age*, in NIBLETT B. (a cura di) *Computer Science and Law: An Advanced Course*, cit., pp. 75-100, ALLEN L.E., *Towards a Normalized Language to Clarify the Structure of Legal Discourse*, in MARTINO A.A. (a cura di), *Deontic Logic, Computational Linguistics and Legal Informations Systems*, cit., pp. 349-407. Si ritornerà sulla normalizzazione nel cap. 8 par. 1.1.

<sup>45</sup>ALLEN L.E., SAXON C.S., *Analysis of the Logical Structure of Legal Rules by a Modernized and Formalized Version of Hohfeld's Fundamental Legal Conceptions*, in MARTINO A.A., SOCCI NATALI F. (a cura di) *Automated Analysis of Legal Texts*, cit., pp. 385-450.

<sup>46</sup>ANDERSON A.R., BELNAP N., *Entailment*, Princeton University Press, Princeton, 1975.

<sup>47</sup>Nell'opera di Allen è costante il riferimento al pensiero di Hohfeld.

<sup>48</sup>ALLEN L.E., SAXON C.S., *Computer Aided Normalizing and Unpacking: some Interesting Machine-processable Transformations of Legal Rules*, in WALTER C. (a cura di), *Computing Power and Legal Reasoning*, cit., pp. 495-572.

programma in PROLOG che può eseguire operazioni inferenziali della logica proposizionale<sup>49</sup>.

La normalizzazione ha esercitato un notevole influsso sulle ricerche informatico-giuridiche: alla normalizzazione si è ispirato il sistema CLLIPS, sopra presentato; tale metodo è stato utilizzato presso l'Istituto di documentazione giuridica di Firenze per alcune ricerche sulla tecnica legislativa<sup>50</sup>; Bing, come vedremo, si è ispirato alla normalizzazione per realizzare un sistema per il reperimento di informazioni concettuale.

Il limite fondamentale delle implementazioni della normalizzazione finora proposte da Allen consiste nel trattamento della sola logica proposizionale: a questo livello né l'analisi logica, né le operazioni inferenziali possono tener conto della struttura subproposizionale del testo. Non sono possibili inferenze di fondamentale importanza nel ragionamento giuridico deduttivo, come il passaggio dal generale all'individuale, o l'effettuazione di operazioni di calcolo numerico.

Questo limite va correlato al fatto che, con la tecnica della normalizzazione, Allen intende predisporre un linguaggio formale per la rappresentazione della conoscenza giuridica direttamente fruibile dal giurista comune (privo di formazione logico-informatica). Un linguaggio di questo tipo non può andare al di là della logica proposizionale, ma, proprio per questa ragione consente solo un trattamento superficiale dei problemi giuridici. I due scopi che Allen si propone con la normalizzazione - cioè (a) l'elaborazione di strumenti linguistici per rendere più rigoroso il linguaggio giuridico e (b) lo sviluppo di basi di conoscenza - non sembrano raggiungibili con gli stessi strumenti. Un'estensione della normalizzazione alla logica predicativa si è sviluppata nel programma IRINORM<sup>51</sup>, ma, come si vedrà, con un certo pregiudizio della comprensibilità del testo.

---

<sup>49</sup>ALLEN L.E., SAXON C.S., *Automatic Generation of a Legal Expert System of Section 7 (2) of the United Kingdom Data Protection Act 1984*, Relazione presentata al 4<sup>o</sup> Congresso internazionale sul tema *Informatica e regolamentazioni giuridiche* (Roma, maggio 1988); ALLEN L.E., SAXON C.S., PAYTON S.A., *Syntesizing Related Rules from Statutes and Cases for Legal Expert Systems*, relazione presentata al Congresso internazionale *Expert Systems in Law* (Bologna, maggio 1989).

<sup>50</sup>MARTINO A.A., *Software per il legislatore*, in "Informatica e diritto", 3, 1987, pp. 25-64.

<sup>51</sup>Cfr. appendice.

## 2.7. Il sistema LIRS

Il progetto LIRS, illustrato da Carole Hafner in una dissertazione presentata all'università del Michigan nel 1978, si proponeva di realizzare un sistema per il reperimento di informazioni concettuale nel settore della disciplina dei titoli di credito<sup>52</sup>.

La Hafner rappresentò circa duecento casi e duecento sottosezioni dell'Uniform Commercial Code in reti semantiche<sup>53</sup> e predispose un linguaggio che le consentiva di ricercare i dati caratterizzati da determinate strutture concettuali.

Ad esempio<sup>54</sup>, per la causa Jackson contro First National Bank of Memphis, si memorizzavano alcune informazioni relative alla posizione delle parti ("l'attore era un traente" e il "convenuto era un trattario"), alla causa dell'azione ("la causa-dell'azione era un assegno tratto impropriamente"), al tipo di causa ("la firma del traente o dell'emittente che era falsificata, e un assegno ritornato al cliente che non era stato esaminato prontamente). Inoltre, si registrava una sintetica analisi giuridica della causa, in un linguaggio formale equivalente all'italiano semplificato qui usato:

SI E' VERIFICATO che c'era un ASSEGNO-IMPROPRIAMENTE-TRATTO;

-a causa dell'ASSEGNO che non era stato NEGOZIATO-CON-RAGIONAVOLE-DILIGENZA;

-e a causa della FIRMA DEL TRAENTE-O-DELL'EMITTENTE che era stata FALSIFICATA;

-e nonostante l'ASSEGNO RITORNATO-AL-CLIENTE che non era stato ESAMINATO-PRONTAMENTE;

-citando come fondamento Uniform Commercial Code 3-406 e Uniform Commercial Code 4-406.

Una volta che i precedenti fossero stati rappresentati in questa forma erano possibili interrogazioni del tipo:

TROVA tutti i casi

-relativi a una GIRATA FALSA,

-nei quali l'ATTORE non era un POSSESSORE-AL-TEMPO-DEBITO

---

<sup>52</sup>HAFNER C.D., *An Information Retrieval System Based on a Computer Model of Legal Knowledge*, UMI Research Press, Ann Arbor (Michigan), 1981; HAFNER C.D., *Representation of Knowledge in a Legal Information Retrieval System*, in ODDY S., ROBERTSON C., VAN RIJSBERGEN C., WILLIAMS P., *Information Retrieval Research*, 1981. -

<sup>53</sup>Sulle reti semantiche, cfr. cap. 7 pag. 4.

<sup>54</sup>Riprendo l'esempio menzionato in MCCARTY T., *Intelligent Legal Information Systems: Problems and Prospects*, cit., pp. 277-278.

-e tali che, se era risultato che la FIRMA era FALSA, allora il CONVENUTO non era un BANCA<sup>55</sup>.

### 3. I principali progetti degli anni 80

Come già si osservava, gli anni 80 dell'informatica giuridica sono stati caratterizzati da un prevalente interesse per le tematiche dell'intelligenza artificiale.

Non sarebbe possibile presentare in modo esauriente tutti i sistemi presentati nel dibattito scientifico. Pertanto ci si limiterà ad illustrare alcuni progetti particolarmente significativi, che assumono un valore paradigmatico rispetto alle principali linee di ricerca. Altri importanti progetti saranno brevemente menzionati nel par. 5.

#### 3.1. La programmazione logica: il progetto PROLOG

La programmazione logica<sup>56</sup> si è presto rivelata uno degli indirizzi più promettenti per le applicazioni giuridiche dell'intelligenza artificiale. A favore di questo approccio vi sono, in generale, i vantaggi correlati all'uso di metodi logici per la rappresentazione della conoscenza<sup>57</sup>. Inoltre, una rappresentazione logica della conoscenza giuridica può vantare importanti precedenti negli studi di logica delle norme, così che risulta facilitato il collegamento tra intelligenza artificiale e cultura filosofico-giuridica.

Come vedremo nel seguito, in linea di principio, qualsiasi applicazione di intelligenza artificiale può essere realizzata con tecniche di programmazione logica<sup>58</sup>. Tuttavia, queste tecniche rendono immediatamente disponibile un approccio determinato: la normativa giuridica può essere

---

<sup>55</sup>Si osservi che la causa sopra descritta non sarebbe reperita con questa interrogazione, in quanto non soddisfa le tutte condizioni specificate nell'interrogazione. Infatti nella causa dell'esempio è falsa la firma dell'emittente e non quella del girante, e il convenuto è una banca.

<sup>56</sup>Abbiamo già introdotto la programmazione logica nel cap. 3 par. 4. Riprenderemo più estesamente questa problematica nel cap. 7 par. 2.

<sup>57</sup>Cfr. cap. 9 par. 1.

<sup>58</sup>Il problema dei limiti e delle prospettive dell'uso della programmazione logica per rappresentare contesti giuridici sarà trattato nel cap. 8 par. 6.



rappresentata come un insieme di assiomi espressi (formalmente) nel linguaggio logico prescelto. Un dimostratore di teoremi deduce le conseguenze che derivano da quegli assiomi e da altre informazioni formalizzate nel sistema (come parti della sua base assiomatica). Ad esempio, i fatti di un caso determinato possono essere anch'essi espressi in un linguaggio logico. Il sistema, unendo la rappresentazione di norme e fatti, disporrà allora di un insieme di premesse sufficienti per dedurre la disciplina del caso particolare<sup>59</sup>.

I risultati più significativi nell'uso della programmazione logica si sono avuti finora con il PROLOG. Non viene usata tutta la logica dei predicati, ma un sottoinsieme determinato: le clausole di Horn estese con la negazione come fallimento. Questo formalismo è infatti un linguaggio semplice, naturale, sufficientemente ricco per molte applicazioni, e per esso sono disponibili dimostratori di teoremi collaudati ed efficienti<sup>60</sup>.

Il progetto PROLOG, avviato presso l'Imperial College di Londra all'inizio degli anni 80, è il primo e il più conosciuto tra i progetti che hanno utilizzato metodi di programmazione logica. Nell'ambito di questo progetto sono state sviluppate applicazioni in diversi settori del diritto inglese: la legislazione in materia di cittadinanza<sup>61</sup> e di immigrazione, la disciplina dei finanziamenti governativi all'industria, aspetti del diritto previdenziale e fiscale<sup>62</sup>. Attualmente è in corso lo sviluppo di un sistema

---

<sup>59</sup>Per una discussione di questo approccio all'intelligenza artificiale nel diritto, cfr. KOWALSKI R.A., SERGOT M.J., *The Use of Logical Models in Legal Problem Solving*, relazione presentata al Congresso internazionale *Expert Systems in Law* (Bologna, maggio 1989).

<sup>60</sup>Per la distinzione tra programmazione logica in senso lato e in senso stretto, cfr. cap. 7 par. 2.

<sup>61</sup>Cfr. SERGOT M.J., SANDRI F., KOWALSKI R.A., KRIWACZEK F., HAMMOND P., CORY H.T., *British Nationality Act as a Logic Programme*, in "Communication of the ACM", 29,5, 1986, pp. 370-386; SERGOT M.J., CORY H.T., HAMMOND P., KOWALSKI R.A., KRIWACZEK F., SANDRI F., *Formalisation of the British Nationality Act*, in ARNOLD C. (a cura di) *Yearbook of Law, Computers and Technology: Volume 2*, Butterworth, London, 1986.

<sup>62</sup>Per più precisi riferimenti a questi lavori, cfr. SERGOT M.J., *The Representation of Law in Computer Programs: A Survey and Comparison of Past and Current Projects*, cit.

che ha per oggetto un ampio settore della disciplina inglese della sicurezza sociale<sup>63</sup>.

Le ricerche sviluppate presso l'Imperial College si caratterizzano per il rigore e la consapevolezza nell'uso della programmazione logica. In particolare, si è posto l'accento sull'uso dichiarativo della programmazione logica: i linguaggi di programmazione logica (e, specificamente, il PROLOG) sono stati usati per ottenere una rappresentazione assiomatica delle conoscenze (norme giuridiche e/o fatti cui debbano essere applicate); è poi un motore inferenziale (lo stesso del sistema PROLOG, o un diverso motore inferenziale), a usare quelle conoscenze, cioè a dedurre dalla rappresentazione assiomatica le conseguenze rilevanti. Gli studiosi partecipanti a questo progetto hanno ripetutamente sottolineato i vantaggi dell'uso dichiarativo della programmazione logica:

(a) Una stessa rappresentazione della conoscenza può essere utilizzata con diversi metodi di deduzione (diversi motori inferenziali).

(b) Come il ragionamento, così anche la spiegazione è una prova logica, più comprensibile che la rappresentazione (il "tracciato") dell'esecuzione di un programma.

(c) Una rappresentazione dichiarativa può essere usata per scopi diversi.

E' possibile anche un uso imperativo del PROLOG che, però non presenta vantaggi decisivi rispetto all'uso di metodi di programmazioni tradizionali<sup>64</sup>. Nel progetto PROLOG

---

<sup>63</sup>Cfr. BENCH-CAPON T.J.M., ROBINSON G.O., ROUTEN T.W., SERGOT M.J., *Logic Programming for Large Scale Applications in Law: A Formalisation of Supplementary Benefit legislation*, in *Proceedings of the First international Conference on Artificial Intelligence and Law* (Boston, Maggio 1987), cit., pp. 190-198.

<sup>64</sup>In questo approccio il PROLOG viene usato, anziché per rappresentare la conoscenza (delegando al motore inferenziale il compito della sua manipolazione), per specificare nel dettaglio le operazioni che il sistema dovrà effettuare nell'interazione con l'utente. Cfr. SERGOT M.J., *The Representation of Law in Computer Programs*, cit. p. 32 s., il quale pone l'accento sulla differenza tra l'approccio dichiarativo seguito presso l'Imperial College e l'approccio imperativo adottato, ad esempio, da SCHLOBOHM A., *TA-- A PROLOG Program which Analyses Income Tax Issues under Section 318 (A) on the Internal Revenue Code*, in WALTER C. (a cura di), *Computing Power and Legal Reasoning*, cit., pp. 765-815. Schlobohm ha realizzato in PROLOG un programma assimilabile a quelli illustrati sopra, nel cap. 2 par. 2.1.2. (nell'articolo citato è riportato l'intero codice del programma).

l'uso dichiarativo della programmazione logica è stato facilitato dall'uso del guscio APES<sup>65</sup>.

Finora, le applicazioni giuridiche della programmazione logica si sono limitate al ragionamento deduttivo. In particolare, nell'ambito del progetto PROLOG, sono stati sviluppati sistemi che deducono conclusioni giuridiche sulla base della una rappresentazione formale (a) della normativa giuridica e (b) del caso da qualificare.

La rappresentazione del caso viene fornita dall'utente nella consultazione del sistema, il quale si limita, a risolvere questioni di diritto<sup>66</sup>. Lo sviluppo rigoroso di modelli logici del diritto ha il merito di aver posto in luce le difficoltà che ineriscono al ragionamento deduttivo nel diritto, difficoltà spesso superficialmente accantonate da chi ha preferito affrontare direttamente aspetti che vanno al di là del ragionamento deduttivo (il ragionamento comune, l'analogia, i concetti indeterminati e così via), seppure con strumenti teorici ed informatici relativamente rozzi.

Tuttavia, sembra che anche presso l'Imperial College, si stiano ora avviando applicazioni che vanno al di là delle esperienze sin qui realizzate. In particolare, si stanno esaminando gli aspetti seguenti: la possibilità di diversi usi di una stessa rappresentazione della conoscenza giuridica (per diversi utenti o con diverse strategie inferenziali), l'impiego nel processo legislativo di rappresentazioni della legislazione in modelli logici (per controllare, ad esempio, se tutte le implicazioni della legislazione progettata siano accettabili), il trattamento di regole incompatibili (espressione, ad esempio, di interpretazioni divergenti della stessa disposizione) e della vaghezza<sup>67</sup>.

---

<sup>65</sup>Su APES, cfr. SERGOT M.J. *A Query-the-User Facility for Logic Programming*, in DEGANO P., SANDEWALL E. (a cura di), *Integrated Interactive Computer Systems*, North-Holland, Amsterdam, 1983; SERGOT M.J., HAMMOND P., *A PROLOG Shell for Logic Based Expert Systems*, in *Expert Systems 83: Proceedings of the 3rd Technical Conference of the British Computer Society Specialist Group on Expert Systems* (Cambridge, Dicembre 1983), British Computer Society, 1983, pp. 94-104; HAMMOND P., SERGOT M.J., *APES. Augmented Prolog for Expert Systems. System Documentation*, Logic Based Systems LTD., 1987 (terza edizione). Alcuni aspetti questo sistema saranno toccati più avanti, nel cap. 8 par. 2.1.1.

<sup>66</sup>Cfr. KOWALSKI R.A., SERGOT M.J., *The Use of Logical Models in Legal Problem Solving*, cit.

<sup>67</sup>Cfr. KOWALSKI R.A., SERGOT M.J., *The Use of Logical Models in Legal Problem Solving*, cit. p. 7 ss.

### 3.2. Il trattamento del linguaggio naturale: il progetto LEX

Il progetto LEX (*Linguistik-und-Logik-basiertes juristisches Expertsystem*) fu avviato nel 1984, grazie ad una collaborazione tra l'università di Tubinga e il centro di ricerca IBM di Heidelberg. Questo progetto si propone di effettuare ricerche, finalizzate allo sviluppo di sistemi informatici, nel campo dell'analisi del discorso, della rappresentazione della conoscenza, dei metodi deduttivi e della formalizzazione del diritto<sup>68</sup>. Come settore applicativo è stato scelto il diritto penale tedesco in materia di disciplina della circolazione stradale.

L'obiettivo principale del progetto LEX è la realizzazione di un sistema di consultazione (un sistema di analisi giuridica) destinato ad un utenza di tipo specialistico, dotata cioè di formazione giuridica. Il sistema dovrebbe aiutare il giurista a sviluppare la propria argomentazione relativa ad un caso determinato, rispondendo alle domande relative alla sussunzione di un caso nella previsione legislativa, la ricerca di casi analoghi, l'individuazione delle circostanze nelle quali è applicabile una certa legge e così via. Inoltre, sempre nell'ambito di LEX, è stato sviluppato anche un sistema per l'istruzione giuridica assistita da elaboratore<sup>69</sup>.

Le decisioni fondamentali nello sviluppo del progetto LEX, sono state le seguenti:

(a) Il dialogo tra l'utente e il sistema doveva svolgersi in linguaggio naturale. Risposte, domande e spiegazioni dovevano essere trattate non come enunciati isolati (come normalmente avviene nei sistemi automatici),

---

<sup>68</sup>Sul sistema LEX, si vedano i contributi compresi in due volumi collettanei, dedicati prevalentemente ad esso o a tematiche correlate ERDMANN U., FIEDLER H., HAFT F., TRAUNMÜLLER R. (cura di), *Computergestützte juristische Expertsysteme*, Attempto Verlag, Tübingen, 1986, pp. 250; in BLAZER A. (a cura di), *Natural Language and the Computer. Scientific Symposium on Syntax and Semantics for Text Processing and Man-Machine-Communication Held on Occasion of the 20th Anniversary of the Science Center Heidelberg of IBM Germany, Heidelberg, FRG, February 25, 1988, Proceedings*, Springer, Berlin, 1988. Per una considerazione degli aspetti linguistici ed informatici, cfr. GUENTHER F., LEHMANN H., SCÖNFELD W., *A Theory for the Representation of Knowledge*, in "IBM Journal of Research and Development", vol. 30, n. 1, January, 1986, pp. 39-56. Per un'aggiornata descrizione del progetto, cfr. LEHMANN H., *The LEX-Project - Concepts and Results*, in BLAZER A. (a cura di), *Natural Language and the Computer*, cit., pp. 112-146.

<sup>69</sup>Si era ritenuto che entrambe le funzioni potessero essere svolte dallo stesso sistema ma, nel corso dello sviluppo del progetto, ci si è resi conto che era necessario sviluppare due sistemi distinti.

ma tenendo conto del contesto del discorso (con una gestione dei riferimenti anaforici, delle ellissi e così via).

(b) Il linguaggio per la rappresentazione della conoscenza doveva avere la potenza e la solidità teorica della logica del primo ordine<sup>70</sup>.

Come formalismo per la rappresentazione della conoscenza fu scelta la teoria della rappresentazione del discorso (*Discourse Representation Theory-DRT*)<sup>71</sup>, che sembrò lo strumento migliore per raggiungere i due obiettivi appena menzionati. La teoria della rappresentazione del discorso è un'estensione della logica predicativa (nel senso che consente di rappresentare anche eventi, stati e contesti temporali) particolarmente adatta a fungere da punto d'arrivo per l'analisi automatica di testi in linguaggio naturale. Nel progetto LEX si è definita una rappresentazione delle modalità deontiche nella teoria della rappresentazione del discorso<sup>72</sup>.

Il sistema LEX comprende un modulo per analizzare il linguaggio naturale, e, in particolare, per generare automaticamente strutture formali della teoria della rappresentazione del discorso a partire da un testo in linguaggio naturale. Il sistema dispone della rappresentazione formale della sintassi e della semantica di un ampio frammento della lingua tedesca, ed è in grado di trattare i riferimenti anaforici e le ellissi grazie ad una considerazione delle situazioni del discorso<sup>73</sup>.

Per quanto attiene al motore inferenziale, dopo un primo prototipo in PROLOG (che usava quindi la procedura di inferenza della risoluzione<sup>74</sup>) si è preferito un diverso approccio, ispirato al metodo dei tableaux, che conduce alla

<sup>70</sup>LEHMANN H., *The LEX-Project - Concepts and Results*, cit. p. 112.

<sup>71</sup>Sulla teoria della rappresentazione del discorso, cfr. cap. 8 par. 2.2.

<sup>72</sup>LEHMANN H., ZOEPFRITZ M., *Formale Behandlung der Begriffs 'Pflicht'*, in HOMMEL G., SCHINDLER S. (a cura di) *GI Jahrestagung 6.-10. Oktober 1986*, II, 1986, pp. 392-405.

<sup>73</sup>Le risposte del sistema non dipendono cioè solo dall'enunciato immediatamente formulato dall'utente, ma da tutti gli enunciati formulati nell'interazione con il sistema, che influenzano l'interpretazione e l'enunciazione di ogni singolo enunciato da parte del sistema stesso. La complessità della problematica mi costringe a limitarmi a qualche breve cenno. Per una più ampia trattazione, cfr. LEHMANN H., *The LEX-Project - Concepts and Results*, cit.

<sup>74</sup>Sul funzionamento del motore inferenziale del PROLOG, cfr. cap. 7 par. 2.3.

dimostrazione mediante la generazione sistematica di contro-esempi<sup>75</sup>. In questo modo sarebbe possibile il trattamento dell'intera logica predicativa con un'efficienza soddisfacente. Più esattamente, è stato sviluppato un algoritmo che consente di trattare in modo uniforme strutture della teoria della rappresentazione del discorso, formule logiche e clausole di Horn.

Il sistema include (come il guscio APES menzionato nel paragrafo precedente) la possibilità di interrogare l'utente su fatti non dimostrabili in base alle premesse di cui dispone. E' prevista, inoltre, una gestione del ragionamento per difetto (default)<sup>76</sup>.

Il progetto LEX si caratterizza rispetto alla gran parte dei progetti informatico-giuridici per l'ambizione degli obiettivi scientifici. Esso si propone di raggiungere risultati di valore generale (soprattutto nel campo del trattamento del linguaggio naturale) partendo dalle applicazioni giuridiche. Le questioni di estrema difficoltà affrontate nell'ambito di questo progetto, questioni per le quali mancano a tutt'oggi soluzioni soddisfacenti, ne fanno una ricerca di estremo interesse scientifico, ma forse lontana da prospettive applicative immediate. Recentemente le ricerche avviate nell'ambito del progetto LEX sono state riprese nel quadro di un più ampio progetto di ricerca sul trattamento informatico del linguaggio naturale, sempre presso il centro di ricerca IBM di Tubinga.

### 3.3. Gli aspetti strategici: i sistemi LSD e SAL

Il sistemi LSD (*Legal Decision-making System*) e SAL (*System for Asbestos Litigation*), sviluppati da D.A. Waterman e M.A. Peterson si propongono determinare il valore di una causa (cioè la somma "equa" che possa essere accettata da entrambe le parti in sede di conciliazione). Questo problema non può essere risolto solo in base alla rappresentazione formale (a) di una interpretazione del diritto vigente e (b) della ricostruzione dei fatti rilevanti), ma richiede la considerazione (c) di tutti gli elementi che concorrono a determinare una previsione attendibile dell'esito del processo, e (d) dell'interesse delle parti ad evitare un processo o ad una sollecita

---

<sup>75</sup>Cfr. GUENTHER F., LEHMANN H., SCÖNFELD W., *A Theory for the Representation of Knowledge*, cit., p. 48 ss.

<sup>76</sup>Cfr. GUENTHER F., LEHMANN H., SCÖNFELD W., *A Theory for the Representation of Knowledge*, cit., p. 48 ss. Si accennerà al ragionamento per difetto (default) nel cap. 5. par. 2.5.

conclusione del processo già iniziato<sup>77</sup>. Sotto questo profilo LSD e SAL differiscono dai sistemi già presentati, i quali si limitano invece ad effettuare analisi giuridiche, cioè a risolvere questioni di diritto a partire da una determinata rappresentazione della normativa e delle fattispecie concrete.

Entrambi i sistemi sono basati su regole e implementati nel guscio ROSIE<sup>78</sup>.

LSD, sviluppato a partire dall'inizio degli anni 80, ha per oggetto la responsabilità civile per la produzione o la vendita di merci difettose<sup>79</sup>, e perviene alla determinazione del valore di una causa in quella materia, considerando una serie di informazioni relative al caso controverso (il tipo di danno causato, la probabilità di provare la responsabilità del produttore, la percentuale della responsabilità di questi, l'abilità degli avvocati, le caratteristiche delle parti) e ulteriori elementi relativi al contesto (come la vicinanza della data del processo). Lo scopo di questo primo programma era teorico: gli autori si proponevano di elaborare un modello da utilizzare in ricerche empiriche sulle conciliazioni stragiudiziali.

Il successivo sistema sviluppato da Waterman e Peterson, SAL<sup>80</sup>, riprende le caratteristiche strutturali e funzionali dei LSD, ma ha uno scopo pratico: aiutare i giuristi a determinare il valore della causa in vista della decisione di iniziare una lite o della determinazione delle condizioni alle quali addivenire ad una conciliazione. SAL ha un oggetto più ristretto: i danni derivati da una malattia,

---

<sup>77</sup>Per realizzare un sistema di questo tipo occorre sviluppare e implementare nel sistema non solo (a) una concezione realistica del diritto (le regole su cui si basa il sistema debbono essere quelle che, di fatto, ispirano i giudici), ma anche (b) un modello empirico della decisione giuridica, sulla traccia del realismo americano e (c) un modello della situazione e degli interessi delle parti.

<sup>78</sup>ROSIE è un linguaggio basato su regole sviluppato per facilitare lo sviluppo di sistemi di conoscenza di notevoli dimensioni. ROSIE è un lontano discendente del sistema MYCIN (cfr. cap. 3 par. 3.1).

<sup>79</sup>WATERMAN D.A., PETERSON M.A., *Rule-based Models of Legal Expertise*, in *Proceedings of the First Annual Conference on Artificial Intelligence* (Stanford (California), Agosto 1980), cit., pp. 272-275; WATERMAN D.A., PETERSON M.A., *Evaluation of Civil Claims: an Expert Systems Approach*, in *"Expert systems"*, 1, 1984, pp. 65-76; WATERMAN D.A., PETERSON M.A., *Models of Legal Decisionmaking*, in *Expert Systems. Techniques, Tools and Applications*, Addison-Wesley, Reading (Mass.), +, pp. 135-185.

<sup>80</sup>WATERMAN D.A., PETERSON M.A., 1986, *Expert System for Legal Decision Making*, in *"Expert systems"*, 3, pp. 212-226

l'asbestosi, ai tecnici che installano isolanti di asbesto. Gli autori affermano che il sistema è concretamente utilizzato per la conciliazione di liti pendenti dinanzi a tribunali statunitensi<sup>81</sup>.

### 3.4. L'uso dei precedenti: il progetto HYPO

Il sistema HYPO, realizzato da K.D. Ashley e E.L. Rissland presso l'università del Massachusetts<sup>82</sup>, si occupa di un importante aspetto del ragionamento giuridico, soprattutto (ma non solamente) nei sistemi di *common law*, la ricerca dei precedenti e il loro uso nell'argomentazione giuridica<sup>83</sup>. Questa problematica è affrontata mediante l'uso di metodi di ragionamento per esempi, una problematica alla quale l'intelligenza artificiale sta dedicando una crescente attenzione<sup>84</sup>. HYPO è stato applicato alla disciplina del segreto commerciale nei settori della tutela del software e del diritto dei contratti.

Il sistema "inizia con l'input di una sintesi dei fatti e quindi:

(1) stabilisce (valuta) la situazione di fatto alla luce di aspetti dottrinali riconosciuti, chiamati "dimensioni";

---

<sup>81</sup>WATERMAN D.A., *New Research Tools to Watch for*, in "California Law Review", 5(3), 1985, pp. 19-21.

<sup>82</sup>ASHLEY K.D., RISSLAND E.L., *Toward Modelling Legal Arguments*, in MARTINO A.A., SOCCI NATALI F. (a cura di) *Automated Analysis of Legal Texts*, cit., pp. 19-39; ASHLEY K.D., RISSLAND E.L., *But, See, Accord. Generating 'Blue Book' Citations in HYPO*, in *Proceedings of the First International Conference on Artificial Intelligence and Law*, cit., pp. 67-74; RISSLAND E.L., ASHLEY K.D., *A Case-based System for Trade Secrets Law*, in *Proceedings of the First International Conference on Artificial Intelligence and Law*, cit., pp. 60-66.

<sup>83</sup>Come abbiamo visto, questo problema era già stato affrontato da Meldman e della Hafner, sempre in riferimento al diritto statunitense.

<sup>84</sup> Il trattamento degli esempi solleva problemi vicini a quelli affrontati da McCarty con l'apparato concettuale di prototipo e trasformazioni. Il sistema HYPO riprende precedenti lavori della Rissland sul ragionamento per esempi, cfr. RISSLAND E.L., SOLOWAY E.M., *Overview of an Example Generation System*, in *Proceedings of the National Conference on Artificial Intelligence (AAAI-80)* (Stanford (California), Agosto 1980); RISSLAND E.L., VALCARCE E.M., ASHLEY K.D., *Explaining and Arguing with Examples*, in *Proceedings of the National Conference on Artificial Intelligence (AAAI-84)*, Austin (Texas), 1984, pp. 288-294



(2) sulla base di questa analisi, reperisce i precedenti rilevanti da una base di conoscenza di casi (*Case Knowledge Base-CKB*);

(3) compara il caso esaminato con altri casi rappresentati nella CKB e, in particolare, distingue il caso esaminato da altri casi le cui conclusioni vanno in senso opposto.

In una misura più limitata

(4) suggerisce argomenti che possono essere adottati e precedenti che possono essere citati per sostenerli;

(5) indica argomenti che possono essere adottati dall'altra parte ed eventuali modi per indebolirli<sup>85</sup>.

Ogni caso è descritto con un *frame*<sup>86</sup>, le cui caselle specificano le caratteristiche delle parti (la qualità di persone fisiche o giuridiche, i prodotti sviluppati, la conoscenza di segreti commerciali ecc.) e informazioni generali sul caso (azioni ed eccezioni fatte valere ecc.). In base alla descrizione di ciascun caso il sistema associa ad esso un'analisi nei termini di certo insieme di dimensioni (*dimensions*), cioè di fattori giuridicamente significativi, suscettibili di determinarne l'esito (tali fattori sono individuati considerando la motivazione di decisioni precedenti). Ad esempio, nel settore del segreto industriale, tra le dimensioni che giocano a favore dell'attore (di chi lamenta una violazione del segreto commerciale) vi sono le seguenti:

-un impiegato è stato pagato una somma considerevole per cambiare datore di lavoro (passando dall'impresa attrice alla convenuta);

-un impiegato ha portato strumenti di produzione (come copie dei programmi o della relativa documentazione) al nuovo datore di lavoro<sup>87</sup>.

Dopo aver associato un insieme di dimensioni applicabili e un insieme di dimensioni quasi applicabili al caso esaminato, il sistema provvede al reperimento dei casi simili (le cui dimensioni si trovino in rapporti determinati con quelle del caso di cui trattasi), e ordina i precedenti reperiti a seconda che siano a favore o contro determinate decisioni.

Il sistema produce quindi lo schema di un'argomentazione, nella quale sono menzionati i precedenti a favore o contro una decisione determinata.

Un interessante aspetto di HYPO è la possibilità di generare casi ipotetici, cioè individuare ulteriori elementi

<sup>85</sup>ASHLEY K.D., RISSLAND E.L., *Toward Modelling Legal Arguments*, p. 21.

<sup>86</sup>Cfr. cap. 7 par. 5.

<sup>87</sup>RISSLAND E.L., *Argument Moves and Hypoteticals*, cit., pp. 129-143.

di fatto potenzialmente rilevanti, o ricostruzioni alternative dei fatti, utilizzabili dall'attore o dall'avversario nello sviluppo delle proprie argomentazioni. Ciò può essere rilevante in quanto, nel diritto, bisogna distinguere tra la verità di un'affermazione e la possibilità di provarla. Il giurista deve considerare tutte le ricostruzioni plausibili dei fatti, e non solo quella che, in ipotesi, sappia corrispondente al vero.

### 3.5. Altri progetti

Nelle pagine precedenti si sono presentate alcune applicazioni giuridiche dell'intelligenza artificiale, così da illustrare le principali direzioni della ricerca in questo settore. Per ovvie ragioni di spazio non si sono potuti esaminare tutti i sistemi finora sviluppati. Tra questi possiamo menzionare brevemente i seguenti:

-Il sistema ABF, sviluppato da J.A. Sprowl, provvede alla redazione di documenti giuridici combinando elementi standardizzati in base alle informazioni fornite dall'utente e alla disciplina giuridica applicabile<sup>88</sup>. Sprowl sta sviluppando un nuovo sistema per la redazione automatica di testi giuridici. Quest'ultimo sistema integra un programma per l'elaborazione di testi e un piccolo sistema di conoscenza che interroga l'utente, accetta le risposte, compone i documenti<sup>89</sup>. A partire dall'esperienza di ABF è stato sviluppato inoltre il sistema OLGS (Office Letter Generation System), per la generazione di lettere d'ufficio, che è in grado estrarre informazioni da databases e di interagire con l'utente in linguaggio seminaturale<sup>90</sup>.

-Il sistema TAXADVISOR è destinato ad una funzione di pianificazione fiscale, cioè ad assistere i fiscalisti nel consigliare ai clienti la migliore organizzazione del loro

---

<sup>88</sup>SPROWL J.A., *Automating the Legal Reasoning Process: A Computer that Uses Regulations and Statutes to Draft Legal Documents*, in "American Bar Foundation Research Journal", 1979, pp. 1-81; SPROWL J.A., *Automated Assembly of Legal Documents*, in NIBLETT B. (a cura di), *Computer Science and Law*, cit., pp. 195-205.

<sup>89</sup>SPROWL J.A., *Assembling Standardized Legal Documents in a Semiautomatic Fashion with a Rule-Based Expert System*, relazione presentata al Congresso internazionale *Expert Systems in Law* (Bologna, maggio 1989).

<sup>90</sup>Cfr. OSMAN M.R., EVENS M., HARR H., SPROWL J., *OLSG-An Office Letter Generation System*, relazione presentata al Congresso internazionale *Expert System in Law* (Bologna, 1989).

patrimonio ai fini di minimizzare le imposte<sup>91</sup>. TAXADVISOR è un sistema basato su regole, sviluppato nel guscio EMYCIN.

-Il sistema ELI, basato su regole di produzione della logica proposizionale, è stato applicato alla normativa in materia di previdenza e assistenza sociale<sup>92</sup>.

-Il sistema per il controllo delle prestazioni sociali realizzato presso la Direction General des Affaires Sanitaires et Sociale d'Aquitaine viene usato per il controllo dei fascicoli relativi alla concessione di un assegno agli adulti handicappati<sup>93</sup>.

-Il sistema BRUITLOG intende offrire una consulenza in materia di diritto amministrativo (in particolare si tratta della normativa in materia di polizia municipale per la tranquillità pubblica e i rumori). Di BRUITLOG sono state sviluppate due implementazioni, l'una in ROSIE e l'altra in PROLOG<sup>94</sup>.

-Presso l'università di Oxford è stato sviluppato un sistema basato su regole di produzione della logica proposizionale, che tratta la materia del diritto di famiglia (in particolare, il divorzio)<sup>95</sup>.

-Un sistema per l'ausilio all'attività giurisdizionale è stato sviluppato presso l'università Notre Dame de la Paix di Namur. Esso è destinato a fungere da supporto per la ricerca documentaria e per il controllo della coerenza del giudizio<sup>96</sup>.

---

<sup>91</sup>MICHAELSEN R.H., *An Expert System for Federal Tax Planning*, in "Expert Systems", 1, 1984, p. 240 ss.

<sup>92</sup>LEITH P., *ELI: An Expert Legislative Consultant*, Relazione presentata al Congresso IEE Conference on Man/Machine Systems (UMIST, Manchester, July 1981); LEITH P., *Hierarchically Structured Production Rules*, in "The Computer Journal", 1983, pp. 26 ss.

<sup>93</sup>LEVINE, MINEL, MEYER, *Un système Expert d'aide au controle de prestations sociales: l'allocation aux adultes handicapés*, Relazione presentata al Congresso Informatique et Droit (Strasburgo, novembre 1987).

<sup>94</sup>BOURCIER D., *Ces systemes dits experts ou comment passer du droit à la decision juridique*, Relazione presentata al Congresso Informatique et Droit (Strasburgo, novembre 1987).

<sup>95</sup>GOLD D.L., SUSSKIND R.E., *Expert Systems in Law: A Jurisprudential and Formal Specification Approach*, in MARTINO A.A., SOCCI NATALI F. (a cura di) *Automated Analysis of Legal Texts*, cit., pp. 625-642; SUSSKIND R.E., *Expert Systems in Law. A Jurisprudential Inquiry*, cit.

<sup>96</sup>Cfr. BODARD F., HELLA M., POULLET Y., STENNE P., *A Prototype ADP System to Assist Judicial Decision Making*, in MARTINO A.A., SOCCI NATALI

-Bing ha promosso il progetto di un sistema per il reperimento concettuale di testi basato sulla normalizzazione di Allen<sup>97</sup>.

-Il sistema OBLOG (*Object-oriented LOGic*), sviluppato presso il centro di ricerca per il diritto dell'informazione (*Forschungsstelle für Informationsrecht*) della società tedesca per la matematica (*Gesellschaft für Mathematik Deutschland-GMD*)<sup>98</sup>, è un sistema per la rappresentazione della conoscenza e la deduzione automatica. OBLOG è stato usato come guscio per sistemi basati sulla conoscenza giuridica destinati alla consultazione e all'insegnamento assistito da elaboratore. Il sistema si caratterizza in quanto (a) cerca di unire alla programmazione logica una rappresentazione di oggetti mediante gerarchie concettuali<sup>99</sup> e (b) mantiene la struttura regola-eccezione<sup>100</sup> anche nella base di conoscenza, grazie ad una gestione del ragionamento per difetto.

-Il progetto Home Expert, realizzato presso l'università del Quebec di Montréal, si propone di realizzare un sistema che offra consulenza ai comuni cittadini in materia di

---

F. (a cura di) *Automated Analysis of Legal Texts*, cit., pp. 187-210; SCHAUSS M., *Prototyp eines Informationssystem zur Hilfe bei der juristischen Entscheidungsfindung*, in ERDMANN U., FIEDLER H., HAFT F., TRAUNMÜLLER R. (cura di), *Computergestützte juristischer Expertensysteme*, cit., pp. 153-164.

<sup>97</sup>BING J., *Designing Text Retrieval Systems for 'Conceptual Searching'*, in *Proceedings of the First International Conference on Artificial Intelligence and Law*, cit., pp. 43-51.

<sup>98</sup>GORDON T.F., *Object-Oriented Predicat Logic and Its Role in Representing Legal Knowledge*, in WALTER C. (a cura di), *Computing Power and Legal Reasoning*, cit., pp. 163-203; GORDON T.F., QUIRCHMAYR G., OBLOG. *Eine Programmiersprache für juristische Expertensysteme*, in ERDMANN U., FIEDLER H., HAFT F., TRAUNMÜLLER R. (cura di), *Computergestützte juristischer Expertensysteme*, cit., pp. 123-134; GORDON T.F., QUIRCHMAYR G., *Ein hybrides Wissensrepräsentationssystem zur Modellierung rechtswissenschaftlicher Probleme*, in 16. Jahrestagung der Gesellschaft für Informatik, Berlin, 1986.

<sup>99</sup>Si colloca cioè, almeno in parte, nella prospettiva di una rappresentazione della conoscenza orientata agli oggetti (*object-oriented*).

<sup>100</sup>Caratteristica del linguaggio giuridico. Il principio *lex specialis derogat generali* consente di introdurre la norma speciale come deroga implicita alla norma generale (nella norma generale non si specifica che essa va applicata solo in quanto non ricorrano le condizioni di applicazione della norma speciale).

locazione. La seconda versione del sistema è stata realizzata con il guscio DEDEREC-expert, scritto in LISP<sup>101</sup>.

-Presso l'IRETIJ (*Institut de Recherche et Etudes pour le Traitement de l'Information Juridique*) dell'Università di Montpellier<sup>102</sup> è stato sviluppato un progetto per la realizzazione di un sistema destinato ad aiutare la decisione giudiziaria in materia di divorzio.

-Il sistema HABLES (*Hesadecimal Arithmetic-Based Interlinguistic Legislative Expert System*) è stato sviluppato dal gruppo di ricerca CALIJ di San Sebastian, diretto da Sanchez-Mazas. HABLES si caratterizza per un approccio di tipo algebrico, e per il tentativo di affrontare aspetti di diritto comparato<sup>103</sup>. In particolare, sono stati rappresentate nel sistema alcune parti dei codici civili italiano, francese, spagnolo e belga.

-Il sistema CHOMEXPERT tratta il diritto canadese in materia di disoccupazione. CHOMEXPERT presenta un particolare interesse per l'approfondimento della complessa problematica della rappresentazione del tempo<sup>104</sup>.

Numerosi altri sistemi, per i quali non v'è lo spazio per un'esame dettagliato, sono stati realizzati grazie a gusci (*shell*) per sistemi esperti su microelaboratore oggi disponibili sul mercato. Questi programmi consentono di realizzare sistemi di conoscenza in tempi brevi, ottenendo risultati spesso iconici ed efficaci (in questi programmi le interfacce sono di regola ben curate), ma molte volte toccano in modo estremamente superficiale i problemi della realizzazione di sistemi basati sulla conoscenza giuridica. Lo sviluppo e l'uso dei sistemi di conoscenza realizzati in

---

<sup>101</sup>Cfr. THOMASSET C., *Expert System in Québec Housing Law: from HOME-Expert I to HOME-Expert II*, Relazione presentata al Congresso internazionale *Expert Systems in Law*, Bologna, 1989.

<sup>102</sup>BILON J.L., *A Knowledge Base of Reasonings and Judgements*, relazione presentata al Congresso internazionale *Expert Systems in Law*, Bologna, 1989.

<sup>103</sup>SANCHEZ-MAZAS M., *Algebraic and Arithmetical Translations on Normative Systems and Applications in Legal Informatics*, in MARTINO A.A. (a cura di) *Deontic Logic, Computational Linguistics and Legal Information Systems*, cit., pp. 169-201; SANCHEZ-MAZAS M., *Un modello aritmetico come sistema esperto per il confronto e l'armonizzazione di legislazioni*, relazione presentata al Congresso internazionale *Expert Systems in Law* (Bologna, maggio 1989).

<sup>104</sup>MACKAAY E., POULIN D., FREMONT J., DENIGER C., BRATLEY P., *The Logic of Time in Law and Legal Expert Systems*, relazione presentata al Congresso internazionale *Expert Systems in Law* (Bologna, maggio 1989).

questo modo potrebbe forse rivelarsi utile soprattutto per la didattica. Tra i gusci più diffusi nell'informatica giuridica, ricordo XI PLUS, un guscio con il quale è stato realizzato un sistema avente ad oggetto la normativa inglese in materia di occupazione<sup>105</sup> e CRYSTAL, utilizzato per l'animazione della disciplina inglese in materia di protezione dei dati<sup>106</sup>.

Negli ultimi anni alcune applicazioni dell'intelligenza artificiale al diritto sono state sviluppate anche nel nostro paese:

-Presso la facoltà di Scienze dell'informazione di Milano, un metainterprete in PROLOG è stato usato per la realizzazione di un sistema sulla normativa universitaria<sup>107</sup>. Il sistema dispone di un meccanismo per l'interrogazione dell'utente sui fatti non dimostrabili. La conoscenza è stata formalizzata in PROLOG (in modo dichiarativo)<sup>108</sup>. Un secondo sistema, più evoluto, è stato applicato alla normativa sul valore aggiunto<sup>109</sup>.

-Il progetto PROLEG<sup>110</sup>, sviluppato presso L'ENIDATA di Bologna, riprende l'esperienza dell'Imperial College<sup>111</sup>. PROLEG usa il guscio APES per trattare la normativa in materia di risparmio energetico e di scarichi inquinanti.

---

<sup>105</sup>KEEN M., MCBRIDE S., *Expert systems for Clarifying Employment Law*, in *Knowledge Based Systems* 86, Online, Pinner, 1986.

<sup>106</sup>SUSSKIND R.E., *Expert Systems in Law and the Data Protection Adviser*, in *"Oxford Journal of Legal Studies"*, 7, 1987, p. 145 ss.

<sup>107</sup>Si trattava di un sistema simile al guscio APES che abbiamo già menzionato più volte.

<sup>108</sup>BERTOCCHI R., *Rappresentazione di testi di leggi in PROLOG*, in *"Informatica e diritto"*, 12, 3, 1986, pp. 155-167.

<sup>109</sup>BERTOCCHI R., FRATTINI F., LANZARONE G.A., MAGLIA A., *Programmazione in logica della legge sull'IVA*, in *Atti del terzo Convegno nazionale sulla Programmazione logica*, 1988, pp. 355-371.

<sup>110</sup>ANDRETTA M., LUGARESI M., ZAMBON F., LOSANO M., NANNINI N., *I linguaggi formali applicati alla rappresentazione di testi normativi: il progetto PROLEG (PROlog applicato alle LEGgi)*, Relazione presentata al 4° Congresso internazionale sul tema *Informatica e regolamentazioni giuridiche* (Roma, maggio 1988); ANDRETTA M., LUGARESI M., ZAMBON F., LOSANO M., NANNINI N., *Uso di linguaggi formali per la rappresentazione di testi normativi: il progetto PROLEG (PROlog applicato alle LEGgi)*, in *Atti del terzo convegno nazionale sulla programmazione logica*, 1988, pp. 373-383.

<sup>111</sup>Cfr. par. 3.1.

PROLEG introduce alcuni interessanti sviluppi tra cui, soprattutto, una rappresentazione dei concetti mediante una rete di frames e un'interfaccia in italiano.

-Alcune esperienze nell'uso della normalizzazione di Allen e nell'uso del PROLOG per la realizzazione di sistemi esperti giuridici sono state realizzate presso l'Istituto per la documentazione giuridica di Firenze<sup>112</sup>. Tali esperienze hanno condotto alla realizzazione del sistema ESPLEX basato su regole di produzione della logica proposizionale e integrato da una rappresentazione di rapporti concettuali. Un interessante aspetto di ESPEX è il l'introduzione nel sistema di alcuni elementi della logica deontica<sup>113</sup>. Presso lo stesso istituto sono stati usati alcuni gusci su microelaboratore e, in particolare, XI PLUS per realizzare un sistema sulla normativa matrimoniale <sup>114</sup>, e CRYSTAL per l'animazione della disciplina normativa ambientale<sup>115</sup>.

-Il sistema Methodus è un programma in BASIC, che opera anch'esso mediante una rappresentazione della conoscenza giuridica al livello della logica proposizionale. Una prima applicazione ha riguardato la legge 298/74 (istituzione dell'albo degli autotrasportatori)<sup>116</sup>.

-Un'interessante applicazione in tema di locazioni commerciali<sup>117</sup> è stata realizzata con ESP/ADVISOR, un guscio basato sul PROLOG.

-Il sistema SEFIT, inteso a fornire agli operatori economici informazioni sull'accesso ai finanziamenti del

---

<sup>112</sup>Cfr., tra i numerosi lavori pubblicati sulle esperienze realizzate presso tale istituto, MARTINO A.A., *Software per il legislatore*, in "Informatica e diritto", 3, 1987, pp. 25-64

<sup>113</sup>In particolare, il sistema comprende assiomi che rendono interdefinibili gli operatori deontici.

<sup>114</sup>CAMELLI, SOCCI NATALI F., *Strumenti automatici nel diritto. Lexis: un progetto di sistema esperto giuridico in XI PLUS*, Relazione presentata al 4° congresso internazionale sul tema *Informatica e regolamentazioni giuridiche* (Roma, maggio 1988).

<sup>115</sup>FAMELI E., NANNUCCI R., *I sistemi esperti nel diritto. Strumenti e metodi di sviluppo*, Relazione presentata al 4° congresso internazionale sul tema *Informatica e regolamentazioni giuridiche* (Roma, maggio 1988).

<sup>116</sup>Cfr. CARIDI G., PELLECCIA S., *Automazione della ricerca giuridica e sistemi esperti*, Angeli, Milano, 1986.

<sup>117</sup>D'AIETTI G., *Un Expert system in tema di locazioni commerciali*, Relazione presentata al 4° Congresso internazionale sul tema *Informatica e regolamentazioni giuridiche* (Roma, maggio 1988).

Fondo per l'innovazione tecnologica è stato realizzato su Machintosh usando il guscio NEXPERT<sup>118</sup>. Il sistema comprende tre sottosistemi (a) il primo per valutare le condizioni di ammissione al finanziamento; (b) il secondo per la scelta del tipo di finanziamento; (c) il terzo per informare l'utente sulle procedure da seguire.

-Il sistema LABEO, alla cui realizzazione ho direttamente partecipato, è un guscio per la realizzazione di sistemi esperti giuridici realizzato in PROLOG. Comprende un insieme di strumenti per formalizzare la conoscenza giuridica nel linguaggio della logica proposizionale, per redigere testi normalizzati, per interrogare la conoscenza così formalizzata<sup>119</sup>.

-Il sistema IRINORM, da me realizzato a partire dall'esperienza di LABEO, consente la formalizzazione della conoscenza giuridica nella logica predicativa, la redazione di corrispondenti testi normalizzati, la generazione e l'interrogazione di basi di conoscenza espresse in regole di produzione della logica predicativa<sup>120</sup>.

-Il progetto IRI, in sviluppo presso l'università di Bologna si propone di realizzare un sistema esperto in materia di impatto ambientale. I metodi della programmazione logica saranno utilizzati (a) per la rappresentazione e l'elaborazione della conoscenza e (b) per la realizzazione

---

<sup>118</sup>VALENTE A., *An Expert System for Technological Innovation*, Relazione presentata al Congresso internazionale *Expert Systems in Law* (Bologna, maggio 1989).

<sup>119</sup>Su LABEO, cfr. SARTOR G., *L'analisi dei testi normativi e l'impiego del programma Labeo*, in AAVV, *La redazione dei testi legislativi: tecniche e strumenti nuovi*, Isas, Palermo, 1986, pp. 27-60; CIAMPI C., *Labeo: un sistema esperto per la rappresentazione e l'elaborazione della conoscenza giuridica*, in AAVV, *La redazione dei testi legislativi: tecniche e strumenti nuovi*, cit., pp. 61-83. Un'ampia descrizione di LABEO è riportata in SARTOR G., *Applicazioni dell'intelligenza artificiale al diritto: il progetto IRI (Irnerio Regioni Italiane)*, Relazione presentata al 4° Congresso internazionale sul tema *Informatica e regolamentazioni giuridiche* (Roma, maggio 1988). Una sintesi è riportata in CIAMPI C., *LABEO: A Knowledge-based Expert System for the <<Animation>> of Legal Texts*, in "Informatica e diritto", 1988, pp. 183-202.

<sup>120</sup>Cfr. SARTOR G., *The Irinorm System: Drafting of Normalized Texts and Generation of Knowledge Bases in Predicate Logic*, relazione presentata al Congresso internazionale *Expert Systems in Law* (Bologna, maggio 1989).



di un'interfaccia in linguaggio naturale<sup>121</sup>. Per quanto concerne l'aspetto sub (a) ci si propone di usare il PROLOG (e, in particolare, il guscio APES) per sviluppare un primo prototipo funzionante, sia di sviluppare un metainterprete in PROLOG in grado di gestire un'estensione delle clausole di Horn (in particolare, si sta considerando il linguaggio per il discorso giuridico proposto da McCarty<sup>122</sup>).

Da ricordare, infine, la realizzazione su microelaboratore di interfacce intelligenti per il collegamento con banche dati giuridiche. Sono stati realizzati sistemi che si propongono anziché l'obiettivo ambizioso del reperimento di informazioni concettuale, lo scopo più limitato di facilitare il collegamento con basi di dati tradizionali e la loro interrogazione. Ricordo il sistema EASYFIND, recentemente sviluppato da UNISYS per l'uso del sistema ITALGIURE, interessante anche perché rappresenta la prima utilizzazione nel diritto (a mia conoscenza) di un nuovo linguaggio dell'intelligenza artificiale, SMALLTALK<sup>123</sup>.

#### 4. Conclusione

Anche da una rassegna incompleta come quella che abbiamo presentato, emergono una molteplicità di approcci diversi. Nelle pagine seguenti si cercherà di predisporre uno schema concettuale per classificare e valutare le applicazioni giuridiche dell'intelligenza artificiale. Si considereranno quindi i possibili usi dei sistemi basati sulla conoscenza giuridica

##### 4.1. Contenuti e prestazioni dei sistemi basati sulla conoscenza giuridica

Si è già illustrata la distinzione tra sistemi di analisi, sistemi di pianificazione e sistemi di reperimento concettuale di informazioni, che classifica i sistemi

---

<sup>121</sup>Cfr. PATTARO E., CASADEI G., SARTOR G., *An Expert System in Environmental Law (IRI Project)*, relazione presentata al Congresso internazionale Expert Systems in Law (Bologna, maggio 1989).

<sup>122</sup>Cfr. cap. 8 par. 2.3.

<sup>123</sup>MARASCHINI F., *L'accesso intelligente alle basi informative*, Relazione presentata al 41 Congresso internazionale sul tema Informatica e regolamentazioni giuridiche (Roma, maggio 1988). SMALLTALK è il più diffuso tra i linguaggi orientati agli oggetti.

informatico giuridici intelligenti in base alle loro prestazioni. Considerando invece i metodi di inferenza e le conoscenze comprese nella base di conoscenza si possono tracciare le seguenti distinzioni.

(a) *Una sola interpretazione del diritto - più interpretazioni configgenti.* Possiamo distinguere tra due tipi di basi di conoscenza giuridica: quelle che comprendono un'unica determinata interpretazione del diritto e quelle che comprendono più ricostruzioni, eventualmente incompatibili dello stesso settore del diritto. Quando le interpretazioni siano incompatibili, il sistema deve gestire informazioni contraddittorie<sup>124</sup>.

(b) *Conoscenze di diritto - conoscenze di fatto.* Possiamo distinguere tra sistemi che contengono solo conoscenza giuridica e sistemi che incorporano anche conoscenze fattuali (in aggiunta alla descrizione del caso) come connessioni causali, regole di esperienza e così via)<sup>125</sup>.

(c) *Ragionamento deduttivo - ragionamento plausibile.* Possiamo distinguere tra sistemi che si limitano al ragionamento deduttivo e sistemi che implementano forme di ragionamento plausibile (probabilistico, analogico ecc.), variamente ispirate al c.d. ragionamento comune<sup>126</sup>.

(d) *Linguaggio formale - linguaggio naturale.* Possiamo distinguere tra sistemi che consentono un'interazione (ai fini della rappresentazione della conoscenza o della consultazione) in un sottoinsieme del linguaggio naturale e sistemi che consentono un'interazione solo in un linguaggio formale<sup>127</sup>.

I sistemi informatico-giuridici, in quanto corrispondano al primo termine di ciascuna delle contrapposizioni che abbiamo presentato, si inquadrano più immediatamente nel

---

<sup>124</sup>Un tentativo di trattare diverse ricostruzioni del medesimo settore del diritto è rappresentato dal sistema HYPO (cfr. par. 3.4). Lo studio del problema della trattazione logica di informazioni giuridiche contraddittorie è stato approfondito anche presso l'Imperial College (cfr. par. 3.1).

<sup>125</sup>Quali esempi di sistemi che comprendono conoscenze empiriche, possiamo ricordare il sistemi LDS e SAL (cfr. par. 3.3).

<sup>126</sup>Tra i sistemi che hanno cercato di simulare il ragionamento analogico, possiamo menzionare TAXMAN II (cfr. par. 2.1) e HYPO.

<sup>127</sup>Lo sviluppo di un sistema informatico giuridico che includa un trattamento del linguaggio naturale (e del linguaggio giuridico) rappresenta l'obiettivo principale del sistema LEX (cfr. par. 3.2). Sembrano includere aspetti di trattamento del linguaggio naturale anche il sistema HYPO e i sistemi sviluppati da Sprowl (cfr. par. 3.5).

paradigma del sistema assiomatico deduttivo, nel senso che si limitano a compiere operazioni deduttive a partire da una rappresentazione formale di contenuti giuridici completa, determinata, univoca, priva di antinomie. In quanto corrispondano al secondo termine, rappresentano, in un certo senso, tentativi di superare quel paradigma.

Negli Stati Uniti d'America l'interesse si è concentrato soprattutto sugli aspetti non deduttivi (sul secondo termine delle contrapposizioni prospettate), mentre in Europa, seppure con importanti eccezioni, si è dedicata maggiore attenzione alla problematica del ragionamento deduttivo.

Questi diversi orientamenti della ricerca si riconnettono, a due aspetti distinti:

(a) il prevalere di diversi orientamenti nell'intelligenza artificiale<sup>128</sup>, cioè di un approccio più pragmatico negli Stati Uniti e di un'impostazione più ispirata alla logica in Europa.

(b) la diversità degli ordinamenti giuridici. La plausibilità (e l'utilità) di una rappresentazione assiomatica del diritto è forse maggiore negli ordinamenti di *civil law* (caratteristici dell'Europa continentale) che negli ordinamenti di *common law* (dei paesi anglosassoni), dove le regole giuridiche debbono essere estratte dai precedenti. Anche nella storia della teoria del diritto americana ed europea possiamo rilevare rispettivamente, come si è spesso osservato, il prevalere di indirizzi antiformalistici (dalla *Sociological Jurisprudence*<sup>129</sup>, al

---

<sup>128</sup> Cfr. cap. 3 paragrafi 3 e 4.

<sup>129</sup> Come è noto, il pensiero giuridico nordamericano fu profondamente influenzato dalla giurisprudenza sociologica, un movimento di pensiero sviluppatosi tra la fine dell'ottocento e l'inizio del novecento (cfr. FASSO' G., *Storia della filosofia del diritto. Volume III. Ottocento e novecento*, Il Mulino, Bologna, 1970, pp. 311 ss). Questo indirizzo, come le scuole di diritto libero europeo, muove dall'esigenza di adattare il diritto ad un mondo sociale in rapida evoluzione e dalla preoccupazione per gli effetti sociali del diritto. La giurisprudenza sociologica è caratterizzato dall'attenzione per il diritto realmente operante (che Pound contrapponeva polemicamente al diritto astratto, che sta solo nei libri), cioè per i comportamenti sociali correlati al diritto, e, in particolare, per il processo di decisione giudiziale. La decisione giudiziale viene collocata all'interno della società, e viene considerata come risposta a problemi sociali piuttosto che come applicazione deduttiva di norme preesistenti. Tra i principali rappresentanti della *Sociological jurisprudence*, ricordiamo O.W Holmes (cui si deve la celebre definizione del diritto "Le previsioni di ciò che i tribunali effettivamente faranno, e nient'altro di più pretenzioso è ciò che io intendo come diritto", HOLMES O.W., *The Path of the Law*, in

realismo giuridico americano<sup>130</sup> fino all'analisi economica del diritto<sup>131</sup>), e di indirizzi più formalistici (dal

---

*Collected Legal Papers*, New York, 1920), R. Pound (che vide nel diritto un'opera di "ingegneria sociale" (*social engeneering*), B.N. Cardozo.

<sup>130</sup>Gli elementi antiformalistici si accentuarono, a partire dagli anni 30, nel c.d. realismo giuridico americano che si caratterizza per il rifiuto del formalismo logico e di ogni forma di concettualismo giuridico" (FASSO' G., *Storia della filosofia del diritto. Volume III. Ottocento e novecento*, cit., pp. 311 ss. Per una breve introduzione al realismo giuridico americano, cfr. TARELLO G., *Realismo giuridico*, in *Novissimo digesto italiano*, XIV, 1968, pp. 923-933). Il realismo giuridico americano mantiene l'interesse per la concretezza della decisione giuridica che aveva caratterizzato la *sociological jurisprudence*, ma insiste sugli aspetti personali della stessa, piuttosto che sulla sua funzione sociale. Gli autori solitamente raccolti in questo indirizzo, si caratterizzano per una concezione empiristica della scienza giuridica. Essi videro nei fatti, e, in particolare nel comportamento dei giudici l'oggetto di questa scienza (non nelle norme o imperativi visti come "dover essere"), e le assegnarono il compito di prevedere quei comportamenti. In questa prospettiva, è utile studiare le formulazioni normative, i precetti, solo nella misura in cui siano efficaci nella determinazione di pratiche (comportamenti sociali futuri) (cfr. TARELLO G., *Realismo giuridico*, in *Novissimo digesto italiano*, ^, p. 931). Alcuni autori, come J. Frank, considerando la difficoltà di rinvenire regolarità empiriche nell'attività giudiziaria, giunsero a negare non solo la possibilità della scienza del diritto, ma anche la certezza del diritto. Secondo questo autore la decisione del giudice è frutto di intuizioni non riconducibili a regole, ed è fondamentalmente imprevedibile, o spiegabile solo con il ricorso a discipline come la psicologia e la psicanalisi. Altri, come K.N. Llewellyn ripresero la definizione del diritto di Holmes (cfr. nota precedente), pur osservando che la funzione di guida delle norme giuridiche è limitata, in quanto il diritto è solo un aspetto dell'ordine sociale.

<sup>131</sup>L'analisi economica del diritto è una corrente del pensiero giuridico nata negli Stati Uniti d'America all'inizio degli anni 60, che si caratterizza per l'applicazione delle teorie microeconomiche neoclassiche ai problemi giuridici. Come osserva KORNHAUSER L.A., *Economique (Analyse -du droit)*, in *Dictionnaire encyclopédique du théorie du droit et de sociologie du droit* (diretto da ARNAUD A.J.), Librairie de droit ed de Jurisprudence, Paris, Story-Scientia, Bruxelles, 1988, p. 129, "l'analisi economica del diritto consiste di quattro tesi indipendenti sul piano logico, ma legate sul piano concettuale. La tesi behaviorista afferma che la teoria microeconomica fornisce una teoria valida per predire il comportamento degli individui in presenza di norme giuridiche. La tesi normativa afferma che i giudici dovrebbero scegliere norme giuridiche efficaci. La tesi descrittiva afferma che le regole giuridiche di *common law* inducono in realtà un comportamento efficace.

positivismo giuridico ottocentesco alla teoria pura del diritto di Kelsen)<sup>132</sup>.

Si tratta, ovviamente, di una distinzione approssimativa. Infatti anche nel pensiero giuridico europeo vi sono state significative tendenze antiformalistiche, che hanno profondamente influenzato la cultura giuridica contemporanea<sup>133</sup>, così come tendenze formalistiche non mancano nella cultura giuridica nordamericana<sup>134</sup>.

---

La tesi evoluzionista afferma che le forze sociali sceglierebbero regole "adattate" o almeno, più efficaci di quelle che avrebbero altrimenti adottato coloro che creano il diritto". Per regola efficace deve intendersi a seconda dei diversi autori, quella che massimizza il prodotto nazionale lordo, o che massimizza l'efficacia di Pareto o altri parametri tratti dalle scienze economiche.

<sup>132</sup>Non è questo il luogo per approfondire l'esame del pensiero kelseniano. Mi limito a rilevare che il punto di partenza di del pensiero di Kelsen è la netta separazione della scienza giuridica sia dalle ideologie politiche e morali, sia dalle scienze empiriche (sociali o naturali (cfr., tra i numerosi scritti di Kelsen che toccano questa problematica Kelsen H., *Reine Rechtslehre*, Verlag Franz Deuticke, Wien, 1960, p. 1, p. 60 s., p. 78 ss.). Un sistema basato sulla conoscenza giuridica, nella prospettiva kelseniana non solo potrebbe, ma dovrebbe prescindere dalle conoscenze socio-economiche. Il sistema kelseniano dovrebbe tuttavia esprimere non una sola interpretazione del diritto, ma tutte le interpretazioni grammaticalmente possibili. Infatti, come è noto, Kelsen attribuisce all'interpretazione "scientifica" del diritto, il compito di rilevare i diversi possibili significati di una disposizione normativa senza operare una scelta tra essi (ivi, p. 352 ss.).

<sup>133</sup>Come, ad esempio, il vasto movimento antiformalistico che diede origine alle varie scuole del diritto libero. E' da osservare, tuttavia, che taluni autori europei di indirizzo realistico, hanno dato alle norme (viste sotto il profilo socio-psicologico) una notevole importanza, come fattori che determinano il comportamento degli operatori giuridici. Così nel realismo giuridico scandinavo e, in particolare nell'opera di A. Ross (per una ricostruzione di questo aspetto del pensiero di Ross, cfr. Pattaro E., *Introduzione al corso di filosofia del diritto*, cit.). Una base di conoscenza giuridica ispirata al pensiero di Ross, non si allontanerebbe notevolmente da una base di conoscenza "kelseniana". Il criterio differenziale non sarebbe il fatto che la base di conoscenza non comprenda norme, ma piuttosto il fatto che essa sarà composta solo dalle norme accolte dagli operatori giuridici.

<sup>134</sup>Basta ricordare il contributo di Hohfeld, i cui principali contributi sono stati raccolti in Hohfeld W.N., *Concetti giuridici fondamentali*, Einaudi, Torino, 1969. Dal pensiero di Hohfeld traggono ispirazione i lavori informatico-giuridici di L.E. Allen.

Un sistema automatico che deduca conclusioni giuridiche da una rappresentazione assiomatico-formale del diritto rappresenta infatti:

(a) un modello immediatamente trasferibile nel paradigma della programmazione logica, e

(b) un modello ispirato ad un approccio in qualche senso "formalistico" o normativistico al diritto. Il diritto è considerato come un sistema di norme, di strutture linguistiche alle quali si applicano regole di deduzione sintatticamente definite per ottenere la disciplina di un caso determinato, facendo astrazione dal contesto socio-politico.

Si tratta di una prospettiva riduttrice, semplificatrice rispetto alla realtà della decisione giuridica. Ciò non significa che non vi siano enormi difficoltà anche per la realizzazione di un sistema informatico giuridico che si limiti al ragionamento deduttivo. Infatti gli attuali sistemi deduttivi automatici sono in grado di gestire in modo soddisfacente solo una parte della logica dei predicati, mentre le deduzioni giuridiche comprendono la logica deontica, la logica temporale, logiche modali, e probabilmente anche logiche di ordine superiore, logiche non monotoniche ecc.

Si noti che il modello del sistema giuridico come sistema assiomatico formale può essere usato anche per implementare concezioni realistiche (in senso epistemologico) del diritto. Quando si assuma che (a) le norme rappresentate nel sistema siano quelle accolte dall'operatore giuridico, (b) la ricostruzione dei fatti da cui muove il sistema coincida con quella cui l'operatore giuridico giungerebbe, (c) questi applichi correttamente i principi logici (e accolga anche le norme deducibili qualora ne accetti le premesse), il modello logico di cui il sistema dispone rappresenta un modello empiricamente adeguato dell'operatore giuridico e quindi del diritto in senso realistico.

Un sistema automatico che prenda in considerazione fattori estranei alle norme (come i sistemi LSD e SAL), o che cerchi di rappresentare il ragionamento analogico (come il progetto TAXMAN), tende a collocarsi, invece,

(a) per quanto attiene agli strumenti informatici nel contesto dell'approccio più pragmatico di cui al cap. ^ precedente,

(b) per quanto attiene all'impostazione teorico giuridica, verso posizioni in qualche senso antiformalistiche. Infatti, l'accento è posto (1) sui fattori extranormativi che condizionano l'attività dell'operatore giuridico (2) sugli aspetti non deduttivi dell'argomentazione giuridica (i quali postulano il riferimento a fattori estranei ad una rappresentazione assiomatica della normativa giuridica).

Tuttavia, si deve evitare una contrapposizione schematica tra i due approcci appena menzionati: gli ultimi sviluppi dell'informatica giuridica sono caratterizzati da importanti contatti tra prospettive apparentemente divergenti. Sono significativi, a questo riguardo, i più recenti sviluppi del pensiero di McCarty. Nel progetto TAXMAN furono usati metodi per la rappresentazione e l'uso della conoscenza giuridica concepiti in termine di creazione e manipolazione di strutture di dati piuttosto che di formalizzazione logica e deduzione automatica. Recentemente, McCarty ha riconosciuto l'esigenza di adottare una rappresentazione della conoscenza di tipo logico, e sta sviluppando di un metainterprete in grado di gestire un'estensione delle clausole di Horn. Questo autore, che aveva dedicato una notevole attenzione a tematiche come il ragionamento analogico e l'apprendimento automatico ha riconosciuto l'esigenza di dare un solido fondamento logico all'uso del ragionamento deduttivo, anche quale premessa per affrontare il problema del trattamento del ragionamento plausibile<sup>135</sup>. Secondo McCarty, un linguaggio per rappresentare la conoscenza giuridica "dovrebbe essere formale: dovrebbe avere una sintassi composizionale, una semantica precisa, un meccanismo di inferenza ben definito"<sup>136</sup>.

---

<sup>135</sup>McCarty illustra in questi termini il rapporto tra la logica e l'approccio da lui seguito in precedenza "alcuni hanno ritenuto che, poiché io uso gli strumenti della logica nella mia formalizzazione, io consideri ora il diritto come un sistema deduttivo, nel quale le risposte a tutti i problemi giuridici possono essere calcolate girando la manovella del macchinario deontico. Invece, io ho coerentemente respinto questa prospettiva in tutti i miei lavori precedenti, e ho anzi cercato di sviluppare una teoria dei concetti indeterminati (*open-textured*) che sono costruiti e modificati nel corso di un'argomentazione giuridica. Tuttavia, vi sono delle procedure deduttive che vengono richiamate nell'ambito di questa teoria, in ruoli subordinati, e queste procedure debbono essere formalizzate correttamente o altrimenti l'intera teoria cadrà (MCCARTY T., *Permission and Obligation: an Informal Introduction*, cit., p. 211)

<sup>136</sup>MCCARTY T., *A Language for Legal Discourse. I Basic Features*, relazione presentata al congresso internazionale *Artificial Intelligence and Legal Reasoning*, Vancouver, 1989, p. 1. McCarty continua come segue "L'interpretazione semantica delle categorie del senso comune dovrebbe essere intuitivamente corretta, cioè dovrebbe generare quelle inferenze che la gente comune (e il giurista comune) genera in situazioni simili. Il meccanismo di inferenza per il linguaggio dovrebbe essere completo e corretto, in linea di principio, ma la completezza e la correttezza potrebbero essere talvolta sacrificate per esigenze di trattabilità

Studiosi orientati verso la programmazione logica (come quelli che collaborano nel progetto PROLOG dell'Imperial College), hanno invece riconosciuto i limiti dell'approccio logico in senso stretto (deduzione delle conseguenze logiche della formulazione assiomatica di un'unica interpretazione del diritto, esattamente definita ed espressa nel linguaggio delle clausole di Horn). Questi studiosi stanno sviluppando tentativi di rappresentare nel medesimo sistema diverse interpretazioni incompatibili e concetti indeterminati<sup>137</sup>, di trattare casi nei quali le informazioni disponibili siano insufficienti, di affrontare la problematica del ragionamento analogico. La rappresentazione formale del ragionamento giuridico richiede, secondo molti autori, l'uso di strutture rappresentative più complesse, e quindi un superamento del linguaggio delle clausole di Horn, pur all'interno di un approccio ispirato alla logica<sup>138</sup>. Anche il più interessante progetto di trattamento del linguaggio giuridico naturale, il summenzionato progetto LEX muove da una prospettiva ispirata alla logica<sup>139</sup>.

#### **4.2. Gli usi dei sistemi basati sulla conoscenza giuridica**

A conclusione dell'esame delle applicazioni dell'intelligenza artificiale nel diritto, saranno riassunte brevemente le funzioni che sistemi basati sulla conoscenza giuridica svolgono, o possono svolgere, nel mondo del diritto.

##### **4.2.1. La soluzione di questioni di diritto**

La maggior parte dei sistemi informatico-giuridici "intelligenti" svolge una funzione di analisi giuridica limitata a questioni di diritto, fornisce una consulenza che consiste, in generale, nel determinare la qualificazione giuridica della situazione descritta dall'utente, nel proporre la soluzione di una questione di diritto. Non vi sono, a mia conoscenza, sistemi specificamente giuridici che si propongano di risolvere anche questioni di fatto (per

---

computazionale, come avviene nel ragionamento umano ordinario (e nel ragionamento giuridico ordinario!)".

<sup>137</sup>Cfr. KOWALSKI R.A., SERGOT M.J., *The Use of Logical Models in Legal Problem Solving*, cit., p. 10 ss.

<sup>138</sup>IL problema dei limiti del linguaggio delle clausole di Horn sarà affrontato ampiamente nel cap. 8 par. 6.

<sup>139</sup>Cfr. GUENTHER F., LEHMANN H., SCÖNFELD W., *A Theory for the Representation of Knowledge*, cit., p. 48.



talune di queste ultime questioni si potranno ovviamente usare sistemi esperti in altre discipline). Nel dialogo con il sistema l'utente specifica la descrizione del caso, e il sistema, applicando le informazioni di cui si compone la sua base di conoscenza, deduce un "effetto giuridico". Come destinatari di questi sistemi si può pensare, da un lato al giurista "generico", privo di una specializzazione nel settore di cui si tratta, dall'altro al cittadino.

Un sistema per l'analisi giuridica è suscettibile di due distinti usi:

(a) Uso decisionale. Il sistema viene usato nel contesto di una decisione giuridica, e propone una soluzione determinata del problema che gli è sottoposto. L'utente si aspetta che il sistema adotti o proponga una decisione "giusta", cioè una decisione conforme ai criteri che egli stesso (o altra persona responsabile) ha introdotto nel sistema. Ad esempio, si pensi ad un sistema che determini se una persona ha diritto ad un certo sussidio, usato da parte dell'amministrazione competente ad erogare il sussidio stesso.

-Uso previsionale. Il sistema viene usato da chi sia interessato a conoscere il diritto in vigore (nel senso del realismo giuridico), cioè il diritto accolto e probabilmente applicato dalle autorità giuridiche<sup>140</sup>. L'utente è interessato a prevedere il comportamento di queste autorità e tale previsione potrà rappresentare la premessa di ulteriori attività e decisioni dell'interessato stesso. Ad esempio, la persona interessata ad ottenere un sussidio può interrogare il sistema (eventualmente il medesimo sistema dell'esempio precedente) per sapere se ha qualche probabilità di ottenere il sussidio, e, quindi, per decidere se presentare la relativa domanda. Questo uso predittivo è giustificato in quanto le decisioni relative siano prese di regola in conformità alla soluzione proposta dal sistema informatico, oppure, all'opposto, in quanto si ritenga che il sistema rispecchi la prassi amministrativa o giudiziale. Nei casi in cui manchi un accordo tra le diverse autorità giuridiche un sistema che si ponga in un approccio predittivo non può sottrarsi all'esigenza di rappresentare (e usare nei propri "ragionamenti") le principali diverse interpretazioni<sup>141</sup>.

E' possibile pensare anche a sistemi che, seppure sulla base di considerazioni giuridiche "pure", alle quali sia

---

<sup>140</sup>Soprattutto dalla magistratura, cui spetta, in ultima istanza, la decisione dei problemi giuridici.

<sup>141</sup>Ciò comporta notevoli problemi logico-informatici, in quanto i problemi del trattamento logico di informazioni contraddittorie sono lontani dall'essere risolti.

estranea ogni valutazione delle realtà sociali e della concretezza dei processi decisionali, svolgano una limitata funzione non solo predittiva ma anche strategica. Questi sistemi potranno fornire la determinazione del comportamento più conveniente nella situazione ideale caratterizzata da quanto segue: accettazione incontrastata della rappresentazione del diritto implementata nel sistema, piena conformità al diritto dei comportamenti dei soggetti di eventuali decisioni giuridiche, assenza di problemi probatori ecc.

#### 4.2.2. Una funzione previsionale o strategica basata (anche) sui fatti

Benché anche i sistemi che contengono solo conoscenze di diritto possano essere usati a fini previsionali, taluni sistemi svolgono una funzione previsionale più ampia. Questi sistemi prendono in considerazione - oltre alle norme e alla logica - una serie di altri fattori, di tipo socio-psicologico, che possono influenzare la decisione giuridica. Ad esempio, nel sistema SAL sono considerate le qualità delle persone coinvolte, la capacità degli avvocati, le difficoltà della prova, la vicinanza del processo, ecc.. Sulla base di questa funzione previsionale alcuni sistemi assumono una valenza strategica: non si limitano ad indicare la probabilità dei diversi esiti di un processo decisionale, ma, sulla base di un'analisi dei costi e dei benefici, valutano diversi comportamenti alternativi. Ad esempio, il sistema SAL indica le condizioni alle quali la conciliazione sia preferibile al processo.

Resta da vedere se, allo stato attuale delle tecnologie informatiche e delle conoscenze sociali, in considerazione della limitata possibilità di identificare regolarità empiricamente osservabili nei comportamenti non giuridicamente regolati degli operatori del diritto, sia ragionevole pensare allo sviluppo di sistemi di questo tipo o se sia preferibile accontentarsi delle più limitate funzioni previsionali o strategiche che possono essere svolte dai sistemi di cui al punto precedente.

#### 4.2.3. La redazione di documenti giuridici

La redazione di documenti è un importante aspetto del lavoro del giurista. I sistemi informatici tradizionali hanno consentito di automatizzare la preparazione di alcuni tipi - o di alcune parti - di documenti giuridici (si pensi alle certificazioni redatte dagli uffici anagrafici, o ad alcune parti standardizzate di un atto notarile, di una citazione, di un'ordinanza, di un precetto ecc.). Dai sistemi di conoscenza ci si attende la redazione automatica anche di documenti da preparare in base alla considerazione

di numerosi fattori e, in particolare, di una normativa giuridica anche complessa, come, ad esempio, contratti, atti amministrativi e giudiziari (non tanto le sentenze quanto piuttosto talune categorie di decreti ed ordinanze) e così via. Inoltre, solo grazie alle tecnologie dell'intelligenza artificiale si potranno realizzare sistemi per la redazione automatica di atti giuridici sufficientemente flessibili da tener conto delle particolari esigenze e interessi di ogni singolo utente.

#### 4.2.4. La ricerca di informazioni giuridiche

Sono stati realizzati sistemi esperti giuridici volti che operano come interfacce intelligente nel campo della ricerca documentaria.

Alcuni sistemi - come, ad esempio, LIRS- si propongono l'obiettivo ambizioso del reperimento di dati concettuale.

L'impiego di sistemi siffatti non pone problemi teorici e etico-politici gravi come quelli che si collegano all'uso dei sistemi di analisi e pianificazione: la ricerca dei documenti mediante sistemi automatici è ormai una realtà accettata dai giuristi.

La realizzazione di banche di dati concettuali di dimensioni sufficienti da risultare interessanti per il giurista presenta invece notevoli difficoltà sul piano della concreta fattibilità. Vi sono due alternative principali per la rappresentazione dei contenuti giuridici nei termini di un modello concettuale:

-Il ricorso ad un analista umano, che provveda a rappresentare, in strutture concettuali, i contenuti giuridici da immettere nella base di dati. Si tratterebbe di un vero salto qualitativo rispetto alle attuali tecniche per la redazione di estratti (*abstract*), salto che postula però sforzi, e soprattutto costi, difficilmente sostenibili.

-Il ricorso ad un sistema automatico per il trattamento del linguaggio naturale che provveda automaticamente alla costruzione delle strutture concettuali a partire dai testi originali (o, eventualmente, da estratti di questi testi, redatti in un adeguato sottoinsieme del linguaggio naturale).

Oggi sembrano maturi i tempi per muoversi in questa seconda direzione. Infatti, al fine dello sviluppo di un sistema per il reperimento concettuale si può esigere una rappresentazione meno approfondita di quella necessaria per un sistema di consultazione, e la gravità delle conseguenze di una formalizzazione sbagliata o addirittura aberrante del testo è molto minore. L'attenzione e le competenze richieste per l'individuazione dei testi rilevanti rispetto ad una determinata questione, non sono paragonabili a quelle che si richiedono per decidere o per prospettare la decisione della stessa questione. Tuttavia, non sono da sottovalutare, le

difficoltà dell'analisi semantica del linguaggio naturale. Finora nessuna esperienza nel campo della redazione automatica di estratti di testi giuridici ha dato risultati operativi.

Accanto ai sistemi per il reperimento di informazioni concettuale, dobbiamo menzionare i sistemi che si propongono l'obiettivo più limitato, ma immediatamente realizzabile, di facilitare l'accesso e l'interrogazione di più basi di dati giuridiche tradizionali<sup>142</sup>. Questi sistemi gestiscono automaticamente il collegamento con diverse banche dati, agevolano l'utente nella scelta della banca dati da interrogare e nell'elaborazione di strategie di ricerca, consentono l'uso di linguaggi personalizzati nell'interrogazione ecc. Si tratta di strumenti realizzabili anche con strumenti informatici tradizionali, ma che le tecniche dell'intelligenza artificiale possono rendere più flessibili ed economici.

#### 4.2.5. L'ausilio nella legislazione

Si è spesso prospettata la possibilità di un uso di sistemi esperti nell'attività legislativa, anche se finora nessuna realizzazione significativa è stata portata a termine. L'aiuto all'attività legislativa è una funzione complessa nella quale dovrebbero integrarsi alcuni dei sistemi e delle prestazioni che abbiamo illustrato. Infatti, a sistemi basati sulla conoscenza potrebbero essere affidati compiti come:

- la "manutenzione" di un corpus normativo. Una volta che la normativa fosse formalizzata nel sistema diverrebbe possibile evidenziare incoerenze e lacune, determinare l'impatto di una nuova disposizione sulla normativa previgente, accantonare le norme abrogate ecc.<sup>143</sup>.

- un sistema informatico può essere usato per valutare alcuni aspetti dell'impatto sociale di una determinata disciplina giuridica<sup>144</sup>.

---

<sup>142</sup>Come il sistema EASYFIND, menzionato nel par. 3.5.

<sup>143</sup>Sull'uso dell'informatica nell'attività legislativo, cfr. OELINGER T. (a cura di), *Gesetzgebung und Computer*, Schweizer Verlag, München, 1981. Cfr. anche MARTINO A.A., *Software per il legislatore*, cit.

<sup>144</sup>Un interessante esempio è il sistema illustrato in STRIPINIS D., *Fiscal Legislation Planner*, in MARTINO A.A., SOCCI NATALI F., *Automated Analysis of Legal Texts*, cit., pp. 629-638. Per un riepilogo di talune esperienze europee cfr. SVOBODA W.R., *Models in Planning Legislation*, in MARTINO A.A., SOCCI NATALI F., *Automated Analysis of Legal Texts*, cit., pp. 831-840. E' da osservare, che, i sistemi informatici destinati alla previsione dell'impatto sociale di una normativa giuridica non sempre

-strumenti informatici possono essere usati per la redazione automatica di testi normativi<sup>145</sup>.

#### 4.2.6. L'apprendimento del diritto

Infine, i sistemi basati sulla conoscenza giuridica possono essere usati a fini didattici, per l'apprendimento del diritto. L'uso del sistema (a condizione che questo disponga di un adeguato meccanismo di spiegazione) può consentire allo studente di ripassare le nozioni giuridiche in esso implementate. I requisiti dei sistemi didattici sono meno elevati di quelli che deve soddisfare un sistema le cui consulenze fungano da premessa di decisioni operative. Pertanto, prototipi di sistemi informatico-giuridici intelligenti potrebbero essere messi a punto e controllati nell'insegnamento del diritto.

Il maggior contributo alla formazione giuridica potrà però venire non tanto dall'uso, quanto dalla preparazione di sistemi basati sulla conoscenza giuridica. L'attività di chiarificazione e di ristrutturazione dei contenuti concettuali necessaria per lo sviluppo di una base di conoscenza giuridica può rappresentare un importante aspetto della formazione del giurista.

---

possono essere qualificati come sistemi basati sulla conoscenza giuridica. Li qualificheremo come tali solo qualora non usino solamente conoscenza economiche, statistiche, ecc. ma anche conoscenze specificamente giuridiche.

<sup>145</sup>Ricordo i numerosi lavori di Allen sull'argomento, e, in particolare ALLEN L.E., SAXON R., *Computer Aided Normalizing and Unpacking: some Interesting Machine-processable Transformations of Legal Rules*, cit. Per un'interessante esperienza, cfr. GRAY G., *Statutes Enacted in Normalized Form: the Legislative Experience in Tennessee*, in WALTER C. (a cura di), *Computing Power and Legal Reasoning*, cit., pp. 621-626.

1. I predecessori (cenni)	67
2. I principali progetti degli anni 70	69
2.1. Il progetto TAXMAN	69
2.2. Il progetto CLLIPS	72
2.3. Il sistema JUDITH	73
2.4. Il progetto di Meldman	73
2.5. Il progetto LEGOL	75
2.6. Logica e diritto nell'opera di L.A. Allen	77
2.7. Il sistema LIRS	80
3. I principali progetti degli anni 80	81
3.1. La programmazione logica: il progetto PROLOG	81
3.2. Il trattamento del linguaggio naturale: il progetto LEX	85
3.3. Gli aspetti strategici: i sistemi LSD e SAL	87
3.4. L'uso dei precedenti: il progetto HYPO	89
3.5. Altri progetti	91
4. Conclusione	98
4.1. Contenuti e prestazioni dei sistemi basati sulla conoscenza giuridica	98
4.2. Gli usi dei sistemi basati sulla conoscenza giuridica	105
4.2.1. La soluzione di questioni di diritto	105
4.2.2. Una funzione previsionale o strategica basata (anche) sui fatti	107
4.2.3. La redazione di documenti giuridici	107
4.2.4. La ricerca di informazioni giuridiche	108
4.2.5. L'ausilio nella legislazione	109
4.2.6. L'apprendimento del diritto	110

## PARTE II

### LA RAPPRESENTAZIONE DELLA CONOSCENZA

La rappresentazione formale della conoscenza è una condizione necessaria per lo sviluppo di applicazioni di intelligenza artificiale<sup>1</sup>, e, in particolare della preparazione di sistemi basati sulla conoscenza giuridica<sup>2</sup>. Nella seconda parte di questo volume si approfondirà questa problematica:

- si esaminerà in generale il tema della rappresentazione formale della conoscenza nell'intelligenza artificiale (cap. 5);

- si considereranno gli specifici problemi della rappresentazione formale della conoscenza giuridica (cap. 6);

- si illustreranno alcuni dei metodi e dei linguaggi per la rappresentazione della conoscenza (cap. 7);

- si presenteranno infine alcune esperienze nella rappresentazione della conoscenza giuridica (cap. 8).

---

<sup>1</sup>Cfr. cap. 1, par. 3.4.2.

<sup>2</sup>Cfr. cap. 2, par. 3.2.





## CAP. 5

# RAPPRESENTAZIONE DELLA CONOSCENZA E INTELLIGENZA ARTIFICIALE

I problemi di cui si occupa l'intelligenza artificiale sono problemi difficili, la cui risoluzione richiede l'uso flessibile di grandi quantità di informazioni, di "conoscenza". La rappresentazione della conoscenza è una delle aree di ricerca principali dell'intelligenza artificiale. Si tratta di sviluppare modelli della conoscenza che consentano a dei sistemi automatici di comportarsi in modo intelligente (di agire in modo appropriato in contesti diversi e non rigidamente predeterminati).

Per rappresentare e comunicare la conoscenza gli uomini usano il loro linguaggio, il linguaggio naturale, che rappresenta uno strumento insostituibile, per la sua versatilità e la sua ricchezza, il suo rapporto con l'esperienza e la cultura.

Il linguaggio naturale è suscettibile di una tecnicizzazione, che lo rende idoneo ad affrontare i particolari problemi che si pongono all'interno di discipline determinate. La tecnicizzazione si limita talvolta all'introduzione di nuovi termini (*termini tecnici*) e alla ridefinizione di termini già in uso (*termini tecnicizzati*), e giunge in determinati casi (come nei contesti matematici, logici, informatici) all'introduzione di linguaggi formali. I linguaggi formali vengono definiti mediante il linguaggio naturale, che funge quindi da metalinguaggio<sup>3</sup> rispetto ad essi e, in questo senso, li ricomprende in sé. Il linguaggio naturale, inteso non come linguaggio comune (linguaggio non tecnico), ma come insieme degli strumenti di cui gli uomini si avvalgono per comunicare con i loro simili, è il linguaggio nel quale vorremmo poter comunicare anche con gli elaboratori. Non esistono però sistemi automatici in grado di usare direttamente conoscenza espressa in linguaggio naturale.

Di qui l'esigenza di una rappresentazione formale della conoscenza. La conoscenza, per poter essere usata da un sistema automatico, va formalizzata in strutture sintattiche

---

<sup>3</sup>Linguaggi formali sono usati da tempo, nella matematica, nella logica, nelle scienze empiriche. L'informatica ha dato luogo alla proliferazione di un'infinità di linguaggi di programmazione, ad una vera babele di linguaggi formali.

rigorose ed uniformi. Solo nel ricorrere di quelle strutture - e non nel significato delle espressioni usate - il sistema può riconoscere la condizione per l'applicazione di determinate regole di ragionamento. Infatti, i sistemi informatici "intelligenti" ragionano operando determinate trasformazioni simboliche, applicando regole di inferenza, o più in generale, regole di trasformazione definite sintatticamente<sup>4</sup>.

La rappresentazione della conoscenza relativa ad un determinato settore dovrebbe possedere, in particolare, le seguenti proprietà:

-adeguatezza rappresentativa: le strutture rappresentative debbono poter rappresentare tutti i tipi di conoscenza necessari in quel settore;

-adeguatezza inferenziale: le strutture rappresentative debbono poter essere manipolate efficientemente dal motore inferenziale in modo da ottenere nuove strutture che corrispondano a nuova conoscenza inferita da quella precedente (dalle strutture rappresentative che compongono la base di conoscenza);

-efficienza nell'acquisizione: le strutture informative debbono consentire di acquisire facilmente nuova informazione, aggiungendola alla precedente<sup>5</sup>.

Anche le conoscenze giuridiche sono rappresentate di regola in linguaggio naturale (nei testi legislativi, nelle sentenze, negli scritti di scienza giuridica). Il linguaggio giuridico è un linguaggio tecnicizzato, accanto ai termini del linguaggio comune, usa termini tecnici<sup>6</sup> o tecnicizzati<sup>7</sup>,

---

<sup>4</sup> Cfr. cap. 1, par. 3.3.

<sup>5</sup>Cfr. RICH E., *Intelligenza artificiale*, cit.

<sup>6</sup>"Per vocaboli (o espressioni) tecnici si intendono quei vocaboli (o quelle espressioni) che non ricorrono nel discorrere abituale dei parlanti quella lingua e non compaiono negli enunciati dell'uso comune, e ricorrono invece solo nel discorrere di chi pratica una data scienza e negli enunciati di quella scienza" (TARELLO G., *L'interpretazione della legge*, cit., p. 108). Ad esempio, i termini "prelazione" e "massa dei creditori" sono termini tecnici delle discipline giuridiche.

<sup>7</sup>"Per vocaboli (o espressioni) tecnicizzati si intendono quei vocaboli dell'uso ordinario che nell'ambito di una scienza o tecnica si specializzano e in questa conservano solo una delle diverse accezioni che hanno nell'uso ordinario, ovvero adottano un senso più ristretto e meglio precisato di quello che hanno nell'uso ordinario" (TARELLO G., *L'interpretazione della legge*, cit., p. 108). Ad esempio, termini come "possesso", "famiglia", "impresa", assumono nel linguaggio giuridico significati speciali (talvolta diversi a seconda del ramo del diritto di cui si tratta).

ma tale tecnicizzazione giunge solo in contesti assai limitati all'uso di un linguaggio formale (per la determinazione di sanzioni, di incentivi o di imposte, il legislatore ricorre talvolta a tabelle e indici numerici). Anche nel mondo del diritto si pone pertanto il problema di una rappresentazione formale della conoscenza dotata delle caratteristiche appena indicate.

## 1. Il problema della formalizzazione

Lo sviluppo di un sistema informatico richiede in ogni caso una formalizzazione, l'uso di un linguaggio più limitato e preciso del linguaggio naturale. E' possibile però distinguere tra programmi tradizionali e sistemi basati sulla conoscenza:

-Nella programmazione tradizionale (algoritmica) la formalizzazione si attua con un'analisi progressiva del compito che il sistema deve svolgere (o, in altri termini, del problema che il sistema deve risolvere). Si passa dalla formulazione di quel compito, alla sua analisi in una sequenza di tappe più semplici, all'analisi di queste in sequenze di procedimenti ancor più semplici finché non si giunge, continuando in questo processo di raffinamento progressivo, ad azioni tanto elementari da corrispondere direttamente da un'istruzione formulata nel linguaggio di programmazione prescelto<sup>8</sup>. La conoscenza necessaria a risolvere un certo problema (o una classe determinata di problemi) si esprime nella specificazione dettagliata del procedimento (l'algoritmo) destinato a risolvere quel problema<sup>9</sup>;

-Nello sviluppo di una base di conoscenza la formalizzazione della conoscenza si attua, invece, mediante (a) un'analisi concettuale che individua l'universo del discorso e specifica i concetti, le relazioni, le strutture rappresentative da utilizzare, e quindi (b) una rappresentazione della conoscenza rilevante e del compito da svolgere nello schema concettuale così individuato. La preparazione di una base della conoscenza può essere pertanto concepita come la costruzione della base assiomatica dalla quale il sistema possa poi inferire (usando eventualmente informazioni ulteriori relative ai casi concreti) le risposte ai problemi che gli vengono sottoposti.

---

<sup>8</sup>Si tratta del metodo *top down*, che abbiamo presentato brevemente nel cap. 1, par. 2.

<sup>9</sup>Cfr. cap. 1, par. 1 e 2.

### 1.3 Il concetto di formalizzazione

Nell'ambito degli studi di intelligenza artificiale sono stati elaborati numerose tecniche per la rappresentazione della conoscenza (alcuni dei quali saranno brevemente illustrate nei cap. 7 e 8). Tutte queste tecniche comportano una formalizzazione della conoscenza, l'uso di un linguaggio formale, cioè di un linguaggio sottoposto a precise regole, "con una sintassi regolare ed invariabile in cui ogni segno possiede una ed una sola funzione"<sup>10</sup>.

Con il termine formalizzazione designiamo sia un'attività (il processo che consiste nell'esprimere un certo contesto in un linguaggio formalizzato), sia il risultato cui conduce quell'attività<sup>11</sup>. Nel nostro contesto possiamo dire che è formalizzazione-attività, il lavoro di chi contribuisce a realizzare la base di conoscenza di un sistema informatico, e che è formalizzazione-risultato, la base di conoscenza così realizzata.

Il concetto di formalizzazione va distinto dai due concetti connessi di algoritmo e di assiomatizzazione:

-Un algoritmo è la descrizione del processo per risolvere un problema che lo riduca ad una procedura effettiva<sup>12</sup>. Il fatto che un problema sia espresso in un linguaggio formale non significa che esista un algoritmo per risolverlo<sup>13</sup>.

<sup>10</sup>PUTNAM H., *Formalizzazione*, in *Enciclopedia Einaudi*, 6, 1979, p. 324.

<sup>11</sup>Questa utile distinzione corrisponde alla distinzione tra interpretazione come attività e come risultato, per la quale, cfr. ROSS A., *Diritto e giustizia*, Einaudi, Torino, 1965 (traduzione di *On Law and Justice*, Steven and Sons, London, 1958), p. 111: "Il tipo di attività che mira a delucidare il significato di un enunciato è detta interpretazione. Questa parola viene anche usata per indicare il risultato di tale attività". I concetti di interpretazione-attività e di interpretazione-risultato sono ripresi e approfonditi da TARELLO G., *L'interpretazione della legge*, cit., pp. 39 ss. L'applicazione di questa distinzione anche alla problematica della formalizzazione è giustificata dal fatto che, come vedremo nel seguito, la formalizzazione altro non è che un tipo di interpretazione, caratterizzato dal fatto che l'interpretazione-risultato, anziché essere formulata in un linguaggio naturale (come avviene nella dottrina giuridica, ma anche nella traduzione tra lingue naturali) è espressa in un linguaggio formale.

<sup>12</sup>Per una definizione del concetto di algoritmo, cfr. cap. 1 par. 1.

<sup>13</sup>Come è noto, K. Gödel, dimostrò che nessun insieme finito e non contraddittorio di assiomi consente di derivare tutte le proposizioni vere di una teoria potente almeno quanto l'aritmetica elementare (Cfr. GÖDEL K., *Über formal unentscheidbare Sätze der Principia Mathematica und*

-Una teoria è assiomatizzata quando in essa si possa distinguere un insieme di enunciati, gli assiomi, dai quali, mediante regole di inferenza determinate, siano deducibili tutti gli enunciati validi della teoria o almeno un insieme significativo di quegli enunciati. Gli assiomi non necessariamente debbono essere espressi in un linguaggio formalizzato. Infatti, si distingue tra sistemi assiomatici formali (ad esempio, la logica del primo ordine) e sistemi assiomatici non formali (ad esempio, la geometria euclidea).

Tuttavia, lo sviluppo di tecniche di formalizzazione si connette storicamente con l'elaborazione di procedure algoritmiche e con la tendenza all'assiomatizzazione<sup>14</sup>:

-Un linguaggio formale è necessario per descrivere un metodo algoritmico e per presentare le premesse della sua applicazione in modo che possa essere automatico il passaggio da quelle premesse all'impiego del metodo stesso. Anche quando non sia disponibile un metodo algoritmico per risolvere tutti i problemi esprimibili in un certo linguaggio (ad esempio, nel linguaggio della logica del primo ordine), possono esservi certi settori nei quali un approccio algoritmico conduce a soluzioni soddisfacenti,

-L'uso di un linguaggio formale consente di esprimere in modo non ambiguo gli assiomi di una teoria e le relative regole di inferenza e, quindi, di valutare la correttezza e la completezza dell'assiomatizzazione.

Pertanto in un senso più stretto, formalizzare una teoria (un insieme deduttivamente chiuso)<sup>15</sup> significa non solo usare un linguaggio formale, ma compiere due operazioni: 1) "formalizzare il linguaggio della teoria", cioè costruire una notazione non ambigua, soggetta a regole, per esprimere gli asserti della teoria; e 2) "formalizzare gli assunti" della teoria, cioè trascrivere gli assiomi (ossia dare una regola effettiva per dire se una formula sia annoverata tra gli assiomi o no), e dare le regole di inferenza (spesso queste sono semplicemente le regole della logica del primo ordine)<sup>16</sup>.

---

*verwandter Systeme I*, in "Monatshefte für Mathematik und Physik", pp. 173-198, 1931 (trad. it. in AGAZZI E., *Introduzione ai problemi dell'assiomatica*, Vita e pensiero, Milano, 1961).

<sup>14</sup>PUTNAM H., *Formalizzazione*, cit., p. 324 ss.

<sup>15</sup>Un insieme deduttivamente chiuso (o un sistema deduttivo), è un insieme di enunciati che comprenda tutte le sue conseguenze: se un certo enunciato è deducibile da un qualsiasi sottoinsieme degli enunciati appartenenti al sistema, anch'esso appartiene al sistema.

<sup>16</sup>PUTNAM H., *Formalizzazione*, cit., pp. 334-335.

Questo concetto di formalizzazione sembra direttamente applicabile più allo sviluppo di un sistema basato sulla conoscenza che ad un programma tradizionale:

- nei programmi tradizionali, in particolare, è utilizzato un linguaggio formale, ma manca una distinzione tra assiomi e regole di inferenza;

- nei sistemi di conoscenza, come si accennava sopra, mediante l'impiego di linguaggi formali si giunge, da un lato alla creazione della base di conoscenza, l'insieme degli assiomi, e dall'altro allo sviluppo del motore inferenziale, che applicando determinate regole di inferenza, deduce ulteriori enunciati del sistema.

#### 1.4 Necessità di una formalizzazione

La tesi che l'elaborazione elettronica dei dati presuppone una formalizzazione, si accorda con l'esperienza di chi abbia usato un linguaggio di programmazione tradizionale o abbia interrogato una banca dati priva di interfaccia in linguaggio naturale. A queste esperienze si riconnette l'idea rassicurante che gli elaboratori elettronici siano macchine "stupide", capaci solamente di eseguire meccanicamente determinate istruzioni elementari. Infatti, i programmi tradizionali sono anche chiamati programmi deterministici: essi consistono nella precisa descrizione dell'algoritmo che l'elaboratore deve eseguire; essi forniscono quindi la soluzione di un unico problema o di un'unica classe ben determinata di problemi.

Questa tesi sembra invece in contrasto con i recenti sviluppi dell'intelligenza artificiale, che hanno condotto alla realizzazione di sistemi che tentano di compiere prestazioni come le seguenti:

- tradurre automaticamente da un linguaggio ad un altro;
- rispondere ad interrogazioni in linguaggio naturale;
- fornire risposte in linguaggio naturale;
- svolgere ragionamenti complessi, anche in contesti caratterizzati dall'incertezza, o dalla carenza di informazioni,
- apprendere dall'esperienza.

Infatti, i sistemi basati sulla conoscenza sono tipicamente basati su programmi "non deterministici" nel senso che è il sistema che, usando un procedimento molto generale (il c.d. meta-algoritmo implementato nel motore inferenziale) determina l'algoritmo da seguire (le operazioni da svolgere per risolvere uno specifico problema). Pertanto, movendo dalla base di conoscenza e applicando le regole di inferenza di cui dispone, il sistema può adattarsi ad un'ampia categoria di problemi.

Bisogna però ricordare che non esistono (né sembrano concepibili, almeno allo stato della tecnologia) sistemi in

grado di "ragionare" direttamente in linguaggio naturale. Anche se il linguaggio nel quale è espressa la base di conoscenza può assomigliare esteriormente ad un determinato linguaggio naturale, si tratta pur sempre di un linguaggio formalizzato, nel quale il significato dei simboli è determinato, anziché dalla sintassi e dalla semantica del linguaggio naturale (e quindi, dalla cultura della quale quel linguaggio è parte), dal modo nel quale i simboli sono usati nella stessa base di conoscenza, e quindi, dalla sintassi (univoca e semplificata) nella quale i simboli sono espressi. Il ragionamento del sistema, anche quando muova da conoscenza "incerte", resta un ragionamento esatto, che consiste nell'eseguire trasformazioni determinate di strutture simboliche.

In conclusione, una rappresentazione formale della conoscenza è la condizione necessaria della realizzazione di un sistema informatico intelligente.

## 1.2 Conoscenze pubbliche e private

Un sistema basato sulla conoscenza dovrà disporre di tutte le informazioni necessarie allo svolgimento del compito al quale è destinato.

Si distingue, a riguardo, tra conoscenze pubbliche, di generale accessibilità (di solito ricavabili dalla letteratura sull'argomento), e conoscenze private, non disponibili al pubblico, soprattutto a causa della loro stretta relazione con concrete esperienze individuali o collettive<sup>17</sup>.

Delle conoscenze private, è depositario l'esperto che svolge i compiti che si intendono affidare al sistema (di volta in volta, il chimico, il medico, l'esperto finanziario, l'avvocato ecc.). Tuttavia, anche le conoscenze pubbliche possono essere adeguatamente padroneggiate solo da chi abbia una specifica formazione in quel settore (il laureato o lo studente in chimica, medicina, economia, giurisprudenza ecc.). Di conseguenza spetta all'esperto (che può essere parzialmente sostituito, per le conoscenze pubbliche, da chi, pur conoscendo il settore, non abbia ancora una sufficiente esperienza) trasmettere questa conoscenza al sistema, eventualmente con l'aiuto del cosiddetto "ingegnere della conoscenza". Per "ingegneria della conoscenza" si intende l'attività consistente nel riformulare la conoscenza al fine del suo uso in un sistema

---

<sup>17</sup>Tra le conoscenze di questo tipo, vi sono la maggior parte delle regole di tipo euristico, che consentono all'esperto di prendere decisioni anche in situazioni caratterizzate dalla mancanza o dall'insufficienza di informazioni.

informatico. Infatti, la conoscenza va strutturata in una rappresentazione che consenta un'efficiente manipolazione automatica, l'aggiornamento e l'incremento della conoscenza stessa, agevoli lo sviluppo di strumenti per correggerne gli errori ed esporre spiegazioni.

Ad un sistema informatico debbono essere trasmesse non solo le informazioni che verrebbero comunicate ad una persona che desidera imparare a svolgere il compito che si intende affidare alla macchina, ma anche ulteriori conoscenze, che restano implicite nella comunicazione umana o che fanno parte della comune esperienza.

A questo riguardo, la distinzione tra conoscenze pubbliche (accessibili a tutti) e private (accessibili al solo esperto) può essere ingannevole, poiché induce l'impressione che per lo sviluppo di un sistema intelligente basti immagazzinare tutte le conoscenze (pubbliche o private) disponibili. Infatti, molte "conoscenze", sia individuali che collettive, non sono né pubbliche né private, ma sono inconse. Queste conoscenze costituiscono il vero limite dell'intelligenza artificiale, limite ben più grave dei costi o delle difficoltà che possono presentarsi per accedere alle conoscenze private, ed è dubbio se e in quale misura si tratti effettivamente di conoscenze, intese come informazioni rappresentabili, comunicabili e tendenzialmente formalizzabili.

Si pensi, ad esempio, alle "conoscenze" linguistiche. Tutti sono in grado di parlare, ma nessuno studioso è stato capace di elaborare una grammatica completa di un linguaggio naturale. Anche se tutti posseggono un'adeguata competenza linguistica, non siamo stati capaci di esplicitarne, in millenni di riflessione filosofica e in secoli di studi scientifici che pochi frammenti. Analoghi problemi si presentano, in generale, nelle attività creative, di fronte alle quali l'automazione, almeno allo stato della tecnica, deve arrestarsi<sup>18</sup>.

Il compito della esplicitazione (e, eventualmente, della riprogettazione) delle strutture fondamentali della conoscenza è un compito tradizionalmente attribuito alla filosofia. Infatti, la mancanza di un'adeguata considerazione della tradizione filosofica ha spesso nociuto agli studi di intelligenza artificiale. Un recupero di tale tradizione, seppure in un quadro concettuale non privo di limiti, si sta ora realizzando nell'ambito della scienza cognitiva, disciplina alla quale, come abbiamo già osservato, collaborano studiosi provenienti da diverse

---

<sup>18</sup>Abbiamo accennato al tema dei limiti dell'informatizzazione e dell'intelligenza artificiale nel cap. 1.



discipline (informatici, matematici, filosofi, psicologi ecc.)<sup>19</sup>.

## 2. Rappresentazione della conoscenza e significati

L'informatica adotta spesso una terminologica antropomorfa, alla quale ci siamo uniformati. Si è parlato della creazione di una base di "conoscenza", da creare trasmettendo la "conoscenza" al sistema "esperto". Occorre precisare che si tratta solamente di metafore: non si tratta della trasmissione di conoscenza nel contesto di una comune cultura. Si tratta dell'espressione della conoscenza in un linguaggio formale e, più specificamente, della realizzazione di strutture formali che possano essere sfruttate da un sistema deduttivo automatico che operi determinate manipolazioni simboliche. Il fatto che tali manipolazioni corrispondano a talune operazioni intellettuali, e che i sistemi cui sia stato fornito tale input siano in grado di svolgere taluni compiti il cui svolgimento da parte dell'uomo richiede intelligenza può servire da spunto per talune riflessioni, ma non ci autorizza ad andare al di là della metafora.

L'esame del concetto di formalizzazione ci consente di specificare meglio che cosa intendiamo per "conoscenza", quando parliamo di un sistema automatico. In un senso ampio, possiamo dire che la conoscenza del sistema risulta da una combinazione di strutture di dati (di informazioni) e di procedure (di metodi per il loro uso), che determinano il comportamento del sistema stesso (in questo senso, la conoscenza di un sistema automatico finisce per coincidere con la sua "intelligenza"<sup>20</sup>).

---

<sup>19</sup>Cfr. cap. 1 par. 3.1.

<sup>20</sup>Infatti, l'intelligenza di un sistema automatico è artificiale, risulta da un insieme di informazioni e metodi precisamente definito. Non vi è l'aspetto della creatività, dell'intuizione, che caratterizza la nostra intelligenza e la rende irriducibile ad un insieme di conoscenze. Si potrebbero sviluppare delle critiche a questa affermazione, in base al fatto che il comportamento di un sistema complesso diventa ben presto difficilmente prevedibile anche per i suoi creatori, che vi sono sistemi in grado di apprendere dall'esperienza ecc. Non mi sembra però che, allo stato della tecnica, questa affermazione possa essere contestata, né che sia auspicabile l'uso di sistemi dotati di un'imprevedibile "creatività" (sul problema cfr. WEIZENBAUM J., *Computer Power and Human Reason. From Judgement to Calculation*, cit., p. 229 ss.).

In un senso più ristretto, ricordando la distinzione tra base di conoscenza e motore inferenziale (la caratteristica strutturale dei sistemi di conoscenza), possiamo chiamare conoscenza l'insieme delle strutture di dati che compongono la base di conoscenza del sistema. Si tratta strutture formali che, in un modello antropomorfo, possiamo interpretare come l'insieme delle proposizioni che il sistema sa, o nelle quali esso crede<sup>21</sup>, e dalle quali esso (il suo motore inferenziale) è in grado di inferire ulteriori informazioni.

Nei capitoli seguenti si considereranno alcuni dei metodi che sono stati proposti per rappresentare la conoscenza e, in particolare, la conoscenza giuridica. Nelle pagine che seguono ci si limiterà a tratteggiare alcune questioni di carattere generale.

## **2.1 Rappresentazione della conoscenza, teorie del significato e strumenti informatici**

Le conoscenze da rappresentare in una base di conoscenza concernono gli oggetti del dominio del discorso (del settore nel quale il sistema deve operare), le loro proprietà e le loro relazioni. Talune relazioni tra oggetti e concetti sono particolarmente importanti e ricorrono in quasi tutti i contesti.

In particolare, vanno ricordate la relazione *è un* (tale relazione si esprime spesso con l'espressione inglese *is a*, o *isa*) e la relazione *è parte di* o *parte di* (*part of*). La relazione *è un* collega il tipo alla specie:

**compravendita è un contratto**  
**contratto è un negozio**  
**cane è un mammifero**

La relazione *parte di* collega un sistema (un fenomeno) composto alle sue parti. Per evitare confusioni con il concetto di parte in senso giuridico, forse è meglio denominare questa relazione con l'espressione *componente di*.

**mano componente di corpo**  
**pagina componente di libro**  
**libro componente di biblioteca**

---

<sup>21</sup>Questa seconda formulazione mi sembra migliore. Infatti si sa solo ciò che è vero, mentre si può credere anche il falso (e anche la base di conoscenza di un sistema informatico può contenere informazioni errate). Inoltre, nel mondo del diritto informatico giuridico, abbiamo a che fare anche con norme e valori, e quindi con proposizioni alle quali, secondo molti studiosi, non si possano attribuire valori di verità.

Queste relazioni presentano un'importante proprietà: la transitività:

per qualsiasi  $X, Y, Z$   
 se  $X \text{ è\_un } Y$  e  $Y \text{ è\_un } Z$ , allora  
 $X \text{ è\_un } Z$ .

Ciò significa che non occorre specificare espressamente tutte le relazioni di questo tipo: una volta che ne abbiamo indicate alcune, altre possono essere dedotte. Ad esempio, dalle seguenti premesse:

contratto è\_un negozio,  
 compravendita è\_un contratto,

possiamo dedurre che

compravendita è\_un negozio.

## 2.2 L'ereditarietà delle proprietà

Un aspetto particolarmente importante delle gerarchie di tipo *è\_un* consiste nella c.d. ereditarietà delle proprietà: possiamo assumere che gli oggetti di un tipo specifico "ereditino" le proprietà del genere cui appartengono.

Ad esempio, posto che i contratti hanno contenuto patrimoniale e che la compravendita è un contratto, possiamo concludere che la compravendita ha contenuto patrimoniale.

Tuttavia, spesso un concetto generale comprende non solo proprietà che appartengono necessariamente a tutti gli individui cui è applicabile quel concetto, ma anche proprietà che si addicono solo alla maggior parte (o, comunque ad una parte significativa) di quegli individui. Possiamo dire ad esempio che le svedesi sono bionde, o che i mammiferi portano nel grembo la prole, e al tempo stesso chiamare svedese anche una ragazza con i capelli neri e chiamare mammiferi anche animali che, come l'ornitorinco, depongono uova. Possiamo definire i contratti come negozi bilaterali (che si concludono con due dichiarazioni che coincidono), e chiamare contratti anche negozi alla cui conclusione basta l'espressione della volontà di una delle parti<sup>22</sup>.

---

<sup>22</sup>Come è noto, una parte importante della dottrina, ritiene che vi siano particolari tipi di contratto (come la mediazione), e particolari modalità di conclusione dei contratti (come la conclusione mediante esecuzione), nei quali basta l'espressione della volontà di una sola parte.

Il problema dell'ereditarietà delle proprietà si presenta anche per l'insieme delle caratteristiche mediante le quali un certo concetto viene definito:

-Se definiamo un concetto indicando le definizioni necessarie e sufficienti per il suo uso, allora ogni individuo cui quel concetto è applicabile, presenta necessariamente tutte le caratteristiche incluse nel concetto. Inoltre, se introduciamo queste definizioni con la tecnica aristotelica della definizione per specie e differenza specifica, allora ogni concetto risulta inserito in una gerarchia determinata e eredita tutte le proprietà mediante le quali sono stati definiti i concetti più generali<sup>23</sup>.

-Se il concetto è definito solo mediante "rassomiglianze di famiglia", non è garantito che le caratteristiche comprese in un concetto generale siano ereditate da tutti gli individui che appartengono al genere, e quindi, neppure da tutte le specie di quel genere. Invece che a condizioni necessarie e sufficienti bisogna fare riferimento ad un prototipo o ad un esempio, e determinare se il caso esaminato si avvicini in misura sufficiente al prototipo.

### 2.3 Concezioni del significato e tecniche definitorie

Più in generale possono contrapporsi tre fondamentali concezioni del significato (e tre corrispondenti soluzioni al problema della definizione)<sup>24</sup>:

"-Classica. Un concetto è definito mediante un genere o supertipo e un insieme di condizioni necessarie e sufficienti che lo differenziano da altre specie dello stesso genere...

-Probabilistica. Un concetto è definito da un insieme di caratteristiche e ogni cosa che presenti una preponderanza di queste caratteristiche è un'istanza del concetto...

-Prototipica. Un concetto è definito da un esempio o prototipo. Un oggetto è un'istanza di un concetto *c* se esso assomiglia al prototipo caratteristico di *c* più che ai prototipi di concetti diversi da *c*".

La tecnica della definizione per condizioni necessarie e sufficienti risale, come è noto, soprattutto ad Aristotele.

---

<sup>23</sup>Alcuni preferiscono parlare di definizioni di termini, anziché di concetti. Qui, senza essere eccessivamente pedanti, parliamo indifferentemente di definizione di concetti e di definizione di termini.

<sup>24</sup>SMITH E.E., MEDIN D.L., *Categories and Concepts*, Freeman, San Francisco (CA), 1981, citato in SOWA J.F., *Conceptual Structures: Information Processing in Mind and Machine*, cit., p. 16 s.

Essa si collega al tentativo di stabilire una diretta relazione tra schemi linguistici e struttura della realtà. Questa tecnica definitoria e la concezione del linguaggio cui si connette, possono essere associate alle filosofie che hanno accompagnato la nascita della logica moderna. Infatti, nei lavori di autori come Russel e Wittgenstein vi è una concezione del linguaggio che si basa sugli assunti seguenti:

- i termini individuali fanno riferimento ad oggetti elementari di cui abbiamo esperienza diretta;

- gli enunciati atomici descrivano fatti elementari, cioè rapporti determinati tra quegli oggetti;

- gli enunciati composti (molecolari) sono riconducibili a combinazioni logiche di proposizioni atomiche<sup>25</sup>.

In questa prospettiva, che esprime la concezione estensionale del significato accolta della semantica della logica classica, il significato di ogni espressione linguistica è riconducibile ad una combinazione, logicamente definita, di rapporti tra oggetti elementari.

Come è noto, questa concezione fu rigettata, in seguito, dallo stesso Wittgenstein<sup>26</sup>, in relazione al suo abbandono dell'atomismo logico e al tentativo di collegare il linguaggio e il significato ad una forma di vita, piuttosto che ad una struttura della realtà. Wittgenstein osserva che vi sono parole non hanno un contenuto fisso applicabile a tutti i casi, ma sono definite solo mediante somiglianze di famiglia in riferimento ad un prototipo<sup>27</sup>.

La concezione probabilistica, fu accolta, ad esempio, da Mill, ed è stata recentemente riproposta dalla teoria della logica incerta (fuzzy)<sup>28</sup>.

<sup>25</sup>Come è noto, questa concezione del significato è formulata in modo efficace e suggestivo nel *Tractatus logico-Philosophicus* di Wittgenstein.

<sup>26</sup>WITTGENSTEIN L., *Philosophische Untersuchungen*, in *Werkausgabe Band 1*, Suhrkamp, Frankfurt, 1984 (1<sup>a</sup> edizione 1952), in particolare n. 65 ss.

<sup>27</sup>Più recentemente autorevoli autori (come KRIPKE S., *Naming and necessity*, in DAVIDSON D., HARMANN G., *Semantics of Natural languages*, Dordrecht, Reidel, 1972, pp. 253-355 e PUTNAM H., *The meaning of Meaning*, in GUNDERSON K. (a cura di) *Language, Mind and Knowledge*, Minneapolis, University of Minnesota Press, 1975, pp. 131-193) hanno contestato che il metodo della definizione per condizioni necessarie e sufficienti sia applicabile ai nomi propri e ai nomi che designano sostanze, specie naturali, grandezze fisiche.

<sup>28</sup>ZADEH L.A., *Fuzzy Logic and its Application to Approximate Reasoning*, in *Information Processing 74*, North Holland, Amsterdam, 1974, pp. 591-594.

## 2.4 Problemi del linguaggio giuridico (cenni)

Nel linguaggio comune sono compresenti i diversi approcci al problema del significato. Così nella definizione delle specie naturali (ad esempio, il concetto di "gatto") concorrono l'elemento prototipico (cui potremmo far riferimento, ad esempio, per determinare se, tra gli animali domestici dei marziani, ve ne sia uno classificabile come gatto) e l'elemento della tassonomia gerarchica (il gatto è un animale, un mammifero, un felino ecc.).

Il linguaggio giuridico, è un linguaggio parzialmente tecnicizzato. Il valore della certezza del diritto e l'esigenza tecnico-legislativa di fare del linguaggio giuridico uno strumento idoneo a conseguire scopi determinati (cioè applicazioni determinate) impongono la ricerca di una certa precisione, il tendenziale passaggio da un approccio prototipico ad un approccio di tipo aristotelico, con l'introduzione di termini tecnici o tecnicizzati mediante definizione (legislativa, giudiziale, dottrinale)<sup>29</sup>. Infatti, se la norma fosse aperta a tutte le possibili interpretazioni (e quindi a tutte le possibili applicazioni) l'intento del legislatore verrebbe probabilmente frustrato.

Tuttavia, si tratta di un obiettivo che è raggiungibile (e il cui raggiungimento è auspicabile) solo entro limiti determinati.

H.L.A. Hart ha illustrato l'incertezza di significato del linguaggio giuridico parlando di struttura aperta del diritto e ricorrendo all'immagine del nucleo chiaro (*clear core*) e della penombra (*penumbra*). Possiamo distinguere tra due possibili situazioni nelle quali si viene a trovare chi debba applicare una disposizione giuridica ad un caso determinato:

(a) Non vi sono incertezze in quanto (a.a) l'espressione linguistica usata dal legislatore è sicuramente applicabile ad un certo oggetto, poiché questo è tipico, è un'istanza di un caso *standard* (di un esempio paradigmatico del concetto), quindi l'oggetto ricade all'interno del nucleo chiaro dell'estensione del concetto; (a.b) l'espressione linguistica sicuramente non è applicabile ad un certo oggetto, poiché questo è estraneo ad ogni esempio prototipico di quel concetto e quindi è sicuramente al di fuori dalla sua estensione;

(b) Sussiste un'incertezza: è dubbio se l'espressione linguistica sia applicabile all'oggetto da considerare poiché sono presenti in questo solo alcuni degli elementi

---

<sup>29</sup>Il problema della definizione nel diritto è stato approfondito da numerosi filosofi e teorici del diritto. Per una recente ricostruzione del dibattito cfr. TARELLO G., *L'interpretazione della legge*, cit., p. 153 ss.

tipici e quindi esso ricade nell'area di penombra che sta ai confini del nucleo chiaro dell'estensione del concetto.

L'incertezza del significato delle prescrizioni giuridiche è una caratteristica inseparabile dalla funzione sociale del diritto. Come osserva lo stesso Hart, vi sono due ostacoli fondamentali, inerenti alla natura umana, che rendono impossibile "regolare in modo non ambiguo e in anticipo qualche sfera di condotta per mezzo di criteri generali che debbono essere usati senza ulteriori direttive ufficiali in occasioni particolari":

- "la nostra relativa non conoscenza dei fatti" e

- "la relativa indeterminatezza dei nostri scopi".

L'incertezza del significato del linguaggio giuridico è in parte riconducibile alla tematica più generale dell'incertezza del significato nel linguaggio naturale, ma presenta specifici problemi. In particolare, nel campo del diritto, le somiglianze di famiglia debbono definirsi più in relazione allo scopo attribuito alla norma da interpretare che in base alle caratteristiche naturalistiche delle situazioni considerate<sup>30</sup>.

Bisogna però osservare che non tutte le incertezze del significato dei testi giuridici sono opportune o necessarie. Talvolta, queste incertezze dipendono dall'esigenza di usare formulazioni sulle quali si possa trovare un compromesso politico, pur in mancanza di una volontà concorde. Più spesso dipendono da una scarsa attenzione all'esigenza della chiarezza del linguaggio giuridico. In questi ultimi casi, l'uso di un linguaggio parzialmente formalizzato potrebbe dare un importante contributo per evitare incertezze inopportune o, comunque, non volute dal legislatore<sup>31</sup>.

Nella ricostruzione dei rapporti tra concetti giuridici, taluni autori in omaggio al rigore della costruzione concettuale, preferiscono adottare la tecnica della definizione per condizioni necessarie e sufficienti. Questo

---

<sup>30</sup>Come esempio, Hart considera una prescrizione che vieti ai veicoli l'accesso ad un parco. Nell'emettere tale prescrizione, il legislatore presumibilmente intendeva far riferimento a casi tipici di veicoli (l'automobile, l'autobus, la motocicletta), tali da turbare la pace del parco, non ad altri oggetti, pur definibili in astratto come veicoli (ad esempio l'automobile giocattolo di un bambino, una bicicletta, dei pattini a rotelle ecc.). Per decidere se anche questi oggetti rappresentino dei veicoli ai sensi di questa prescrizione si dovrà considerare lo scopo della norma, scopo che, d'altronde, è relativamente indeterminato finché il caso non si sia presentato.

<sup>31</sup>Questo problema è stato particolarmente approfondito da L. E. Allen, che, come vedremo (cap. 8 par. 1.1), ha elaborato una tecnica redazionale (la normalizzazione), che usa la sintassi logica per evitare incertezze non volute nel linguaggio dei testi giuridici.

approccio può comportare la necessità di allontanarsi dal linguaggio del legislatore, o di ricorrere a taluni artifici concettuali.

Altri invece, per mantenersi più vicini al linguaggio comune e al linguaggio legislativo, preferiscono adottare definizioni più flessibili (che indichino requisiti di massima presenti, ma nessuno dei quali è necessario o sufficiente affinché sia applicabile il termine definito). Questo secondo approccio comporta invece una parziale rinuncia del giurista al compito di rendere chiaro e rigoroso il linguaggio giuridico.

Ad esempio, possiamo definire il concetto di contratto nei termini delle condizioni necessarie e sufficienti affinché un fenomeno si qualifichi come tale, ad esempio come "negozio plurilaterale a contenuto patrimoniale"<sup>32</sup>; o proporre invece una definizione più articolata, che comprende taluni requisiti che di regola sono presenti, ed ammettere tuttavia che, anche in assenza di ciascuno di questi requisiti, in determinate ipotesi un atto possa essere qualificato come un contratto<sup>33</sup>.

## **2.5 Conclusione: teorie del significato e strumenti informatici**

Sono stati sviluppati metodi informatici ispirati a ciascuna delle teorie del significato e della definizione che abbiamo illustrato nelle pagine precedenti.

Tuttavia il metodo aristotelico della definizione mediante condizioni necessarie e sufficienti e, in misura minore, il metodo probabilistico, si prestano ad una più facile implementazione in un sistema informatico. Si tratta infatti di un approccio che consente di trattare le definizioni dei rapporti concettuali come parte di una base assiomatica formulata in linguaggio logico-matematico. Questa base assiomatica può fungere da premessa di inferenze deduttive e, pertanto, è suscettibile di un trattamento informatico mediante strumenti consolidati (che si

---

<sup>32</sup>Cfr. SANTORO PASSARELLI F., *Dottrine generali del diritto civile*, Jovene, Napoli, 1966, p. 221 s.

<sup>33</sup>Cfr. SACCO R., *La nozione del contratto*, in *Trattato di diritto privato, 10 Obbligazioni e contratti*, Tomo secondo, UTET, Torino, p. 12 s. il quale osserva che, l'unica soluzione del problema definitorio del concetto di contratto consiste "nell'adottare una definizione di base, sottostante al nostro diritto vigente, la quale unisca alla semplicità un certo grado di vicinanza al diritto positivo; e poi farla seguire da un numero più o meno altro di correzioni e regole che riducano via via lo scarto tra definizione e norma fino ad eliminarlo".



inquadraano immediatamente nel paradigma del metodo assiomatico-deduttivo).

L'uso del metodo della definizione prototipica presenta invece notevoli difficoltà. Se un concetto è definito mediante un prototipo, e un oggetto non possiede tutte le caratteristiche del prototipo, per risolvere il problema dell'applicazione del concetto a quell'oggetto (della sussunzione della fattispecie concreta nella fattispecie astratta, direbbe un giurista) bisognerà ricorrere a metodi non deduttivi (a forme di ragionamento analogico). Nessun sistema automatico è in grado di trattare soddisfacentemente questi metodi ed il loro uso in sistemi automatici, come si è osservato, richiede uno stretto controllo per essere accettabile nei contesti giuridici.

Per trattare adeguatamente il problema dell'ereditarietà delle proprietà per i concetti che comprendono caratteristiche che non si addicono necessariamente a tutti gli individui cui sono applicabili, bisogna ricorrere a forme di ragionamento per default<sup>34</sup>. Bisogna muovere dall'assunto che ogni specie acquisti tutte le caratteristiche del genere cui appartiene, ma lasciare cadere questo assunto qualora siano indicate per la specie qualità diverse da quelle del genere. Ciò presuppone tuttavia, che si sia già determinato se un certo oggetto rientra nella classe cui un termine fa riferimento. Quest'ultimo problema, come si è rilevato, non può essere risolto con metodi deduttivi.

---

<sup>34</sup>Il ragionamento per difetto si caratterizza per il fatto di muovere da determinati assunti (defaults), accolti dal sistema in certe condizioni che sono destinati a cadere se siano in contrasto con altre informazioni di cui il sistema dispone. Ad esempio, per riprendere un celebre esempio, posto che Tweety è un uccello, posso avanzare l'ipotesi (accogliere il default) che Tweety voli, ma se scopro che Tweety è uno struzzo, poiché so che gli struzzi non volano, dovrò lasciar cadere quell'ipotesi. Sul ragionamento per difetto cfr. POOLE D., *A Logical Framework for Default Reasoning*, in "Artificial Intelligence", 36, 1988, pp. 27-47.

1. Il problema della formalizzazione	114
1.3 Il concetto di formalizzazione	115
1.4 Necessità di una formalizzazione	117
1.2 Conoscenze pubbliche e private	118
2. Rappresentazione della conoscenza e significati	120
2.1 Rappresentazione della conoscenza, teorie del significato e strumenti informatici	121
2.2 L'ereditarietà delle proprietà	122
2.3 Concezioni del significato e tecniche definitorie	123
2.4 Problemi del linguaggio giuridico (cenni)	125
2.5 Conclusione: teorie del significato e strumenti informatici	127

## CAP. 6

# RAPPRESENTAZIONE E FORMALIZZAZIONE DELLA CONOSCENZA GIURIDICA

Anche lo sviluppo di sistemi informatico-giuridici "intelligenti", e, in particolare, la costruzione di sistemi basati sulla conoscenza giuridica, presuppone una rappresentazione formale della conoscenza. In particolare, una formalizzazione del diritto comprenderà:

(a) l'impiego di un linguaggio formale per rappresentare un determinato settore del diritto;

(b) l'individuazione di un insieme di regole di inferenza e di un insieme di enunciati (formulati in quel linguaggio formale) tali che, applicando le regole di inferenza a detti enunciati si possano ottenere tutte le conseguenze rilevanti per il settore giuridico di cui trattasi, o almeno un insieme significativo di tali conseguenze.

Si tratta di realizzare una rappresentazione del diritto come sistema assiomatico formale. Pertanto, la problematica informatico-giuridica dei sistemi basati sulla conoscenza può essere collegata al secolare dibattito filosofico-giuridico sulla possibilità, i limiti, le prospettive di una "assiomatizzazione" del diritto<sup>1</sup>.

Come vi vedrà, la formalizzazione (l'assiomatizzazione) del diritto solleva problemi di enorme complessità e presenta aspetti peculiari rispetto ai temi generali esaminati nel capitolo precedente.

### 1. L'assiomatizzazione del diritto

L'esigenza di una rappresentazione assiomatica del diritto veniva posta con chiarezza già da Leibniz<sup>2</sup>. Questo

---

<sup>1</sup>Sul problema dell'assiomatizzazione del diritto, cfr. RÖDIG J., *Axiomatisierbarkeit juristischer Systeme*, in RÖDIG J., *Schriften zur Juristischen Logik* (a cura di BUND E., SCHMIEDEL G., THIELER-MEVISSSEN G.), 1980, pp. 40-90 (già in KAUFMANN A. (a cura di) *Münchener Ringvorlesung EDV und Recht. Möglichkeiten und Probleme*, in "EDV und Recht", 6, 1973).

<sup>2</sup>Pensatore al quale si debbono fondamentali intuizioni nelle discipline vicine all'informatica giuridica.

Infatti, come è noto Leibniz costruì una macchina calcolatrice e propose un calcolo dei concetti, un metodo per rendere meccaniche talune

pensatore univa infatti una concezione del diritto come sistema assiomatico (come insieme di proposizioni che comprende tutte le sue conseguenze), alla proposta di un metodo ispirato alla logica e alla matematica, per la legislazione e per l'esposizione scientifica del diritto. Il diritto doveva essere introdotto come una base assiomatica, definendo con precisione un insieme di proposizioni che consentissero di dedurre la disciplina giuridica di ogni singolo caso, ma non comprendessero proposizioni ridondanti (già deducibili nel sistema)<sup>3</sup>.

Leibniz si poneva così all'inizio di una scuola che, come osserva G. Tarello<sup>4</sup> "condiziona la cultura giuridica tedesca dell'ottocento, dalla c.d. scuola filosofica alla c.d. Pandettistica"<sup>5</sup>. Tale scuola "ha elaborato una concezione del diritto come sistema di proposizioni; proposizioni che connettono a soggetti (giuridici), predicati (giuridici); proposizioni che sono vere, e dalle quali, con regole di trasformazioni delle proposizioni che troviamo nella scienza logica, possiamo pervenire ad altre proposizioni (giuridiche) vere, e incluse nel sistema giuridico".

Tuttavia, anche se l'assiomatizzazione del diritto può collegarsi storicamente all'approccio che culmina nella c.d. giurisprudenza dei concetti (*Begriffsjurisprudenz*), le due prospettive non coincidono completamente. In particolare, l'idea di una deduzione "produttiva" è estranea ad una corretta applicazione del metodo assiomatico. Come osserva

---

le operazioni del pensiero. Norbert Wiener, l'inventore della cibernetica, vide il germe dell'informatica nel calcolo dei concetti proposto da Leibniz.

E' meno conosciuto il contributo di Leibniz alla logica deontica. Leibniz per primo individuò con precisione i concetti deontici e ne specificò i rapporti con i concetti modali, anticipando acquisizioni del nostro secolo (cfr. KALINOWSKI G., GARDIES J.L., *Un logicien déontique avant la lettre: Gottfried Wilhelm Leibniz*, in ARSP, LX/1, 1974, pp. 79-112).

<sup>3</sup>Il tema, trattato in numerose opere giuridiche di questo autore, è approfondito soprattutto in LEIBNIZ G.W., *Nova methodus docendae discendaeque jurisprudentiae ex artis didacicae principiis in parte generali premissis experientiae luce* (1667), in *Sämtliche Schriften und Briefe*, vol. VI, I, Darmstadt, 1930, pp. 259-634.

<sup>4</sup>TARELLO G., *Storia della cultura giuridica moderna I. Assolutismo e codificazione del diritto*, Il Mulino, Bologna, 1976, p. 133.

<sup>5</sup>E quindi tutta la cultura giuridica moderna, soprattutto dell'Europa continentale.

J. Rödig<sup>6</sup>, le deduzioni di enunciati normativi dai concetti (metodo caratteristico della giurisprudenza dei concetti) non sono deduzioni da un sistema assiomatico (da un insieme definito di enunciati), ma piuttosto da "un insieme di enunciati non definiti, e spesso anche non discussi, associati al concetto di cui trattasi"<sup>7</sup>. Proprio il metodo assiomatico, che costringe a specificare e a precisare le premesse dell'argomentazione e le regole di inferenza, impedisce questo tipo di concettualismo.

Recentemente il problema della assiomatizzabilità del diritto, si è riproposto nell'ambito del dibattito tra posizioni formaliste e antiformaliste nel diritto, e in particolare, nel contrasto tra i fautori dell'applicazione della logica formale al diritto e coloro che invece ritengono che nel diritto si richiedano strumenti di pensiero diversi e incompatibili rispetto al metodo deduttivo (la logica non formale, contenutistica, la topica, la retorica ecc.)

A mio parere questo conflitto può essere superato sulla base delle seguenti considerazioni:

(a) Bisogna distinguere tra contesto della scoperta e contesto della giustificazione. L'esigenza che una conclusione giuridica venga giustificata in un contesto deduttivo, non significa che si debba essere giunti, di fatto, a tale conclusione applicando metodi deduttivi a premesse predeterminate. Né significa che quelle premesse debbano essere mantenute quando siano inaccettabili le conclusioni che ne derivano.

(b) Il significato dell'uso di un linguaggio formale nel diritto dipende dal contesto e dagli scopi del suo utilizzo. In particolare, vi è una profonda differenza tra l'impiego di un linguaggio formalizzato direttamente da parte del legislatore, e il suo impiego per riformulare ciò che è stato espresso in linguaggio naturale con un precedente atto normativo, e che, normalmente, risulta da qualche documento ufficiale. Nel primo caso si possono effettivamente superare talune delle incertezze dell'interpretazione e dell'applicazione del diritto. Nel secondo, invece, con la redazione del testo formalizzato corrispondente ad un dato documento in linguaggio naturale, non si fa che proporre una

---

<sup>6</sup>RÖDIG J., *Axiomatisierbarkeit juristischer Systeme*, cit., p. 56-57.

<sup>7</sup>Ad esempio, come rileva lo stesso Rödig, la c.d. deduzione dell'inammissibilità di contratti di diritto pubblico dal concetto di contratto, è presentata dalla dogmatica giuridica della fine dell'ottocento in modo apodittico, ma si basa su premesse ulteriori non esplicitate e, pertanto, sottratte alla discussione (in particolare, la premessa che nel diritto pubblico non vi siano almeno due soggetti dotati della medesima personalità giuridica).

particolare interpretazione di quel documento (o, eventualmente, più interpretazioni alternative)<sup>8</sup>. Potremmo distinguere corrispondentemente tra (a) sistemi basati su una rappresentazione formale della conoscenza giuridica che risulti direttamente dal linguaggio legislativo (sistemi le cui conclusioni sarebbero incontrovertibili) e (b) sistemi basati invece su una formalizzazione che risulti da operazioni interpretative. La prima ipotesi è completamente teorica: ragioni non contingenti<sup>9</sup> impediscono che il legislatore usi un linguaggio così formalizzato da rappresentare la base di conoscenza di un sistema automatico<sup>10</sup>. Un modello assiomatico del diritto rappresenta non il punto di partenza, ma il punto d'arrivo dell'attività interpretativa<sup>11</sup>.

---

<sup>8</sup>L'impiego di un linguaggio formalizzato potrà però consentire di formulare con maggiore precisione l'interpretazione o le interpretazioni di cui si tratta.

<sup>9</sup>Abbiamo considerato alcuni aspetti del problema nel cap. 5 par. 2.4.

<sup>10</sup>T. Viehweg descrive così un diritto ridotto a sistema assiomatico: "Una rigida assiomatizzazione dell'intero diritto si collega con un rigido divieto di interpretazione all'interno del sistema, il che dovrebbe realizzarsi nel modo più completo attraverso l'uso del calcolo (*Kalkülisierung*); direttive interpretative rigide ed orientate esclusivamente al sistema giuridico (o al calcolo giuridico) per il fatto; ammissibilità illimitata di decisioni di *non liquet*; un'accresciuta intromissione di un legislatore sistematico (o calcolatore) che operi in modo esatto affinché - senza disturbare la perfezione logica del sistema (o del calcolo) - siano resi risolvibili in un secondo momento i casi nuovi, dapprima non risolvibili " (VIEHWEG T., *Topik und Jurisprudenz. Ein Beitrag zur rechtswissenschaftlichen Grundlagenforschung*, cit., p. 62). Nella situazione prospettata da Viehweg, il legislatore non solo adotta un linguaggio formale (ad esempio, il linguaggio della logica), ma formalizza anche i rapporti tra i concetti (i predicati), riconducendoli a pochi concetti primitivi, definibili in modo non ambiguo. Anche se un diritto di questo tipo è realizzabile (cfr. cap. 5 par. 2.4) non bisogna dimenticare, tuttavia, il rischio che l'uso di sistemi informatici intelligenti induca ad un adeguamento dell'uomo alla macchina, e quindi ad un rigido formalismo, che presenterebbe alcuni aspetti della prospettiva delineata da Viehweg..

<sup>11</sup>A mio parere, come illustrerò nel seguito, nell'applicazione del diritto assistita da un sistema basato sulla conoscenza, le istanze che stanno alla base delle posizioni antiformalistiche possono essere fatte valere proprio nell'interpretazione (intesa, in senso ampio, come l'attività intesa a ricostruire il diritto da applicare) quale premessa dello sviluppo di un modello deduttivo corrispondente.

(c) Non necessariamente uomini ed elaboratori elettronici debbono adottare gli stessi metodi di ragionamento. In particolare, l'impiego del metodo deduttivo per "scoprire" le conseguenze di un insieme di assiomi dato può trovare un'ampia applicazione in sistemi automatici, anche se l'uomo è solito limitare le operazioni deduttive al contesto della giustificazione.

L'uso di sistemi basati sulla conoscenza giuridica (e, in particolare, proprio di sistemi ispirati al rigore deduttivo della logica) non necessariamente impedisce di conciliare le due tendenze che da sempre si fronteggiano nel pensiero filosofico giuridico:

(a) i valori della certezza del diritto e dell'uguaglianza, e quindi l'esigenza che ogni decisione giuridica sia fondata su una rigorosa (oggettiva) deduzione a partire da premesse applicabili universalmente;

(b) l'esigenza di tener conto degli interessi concreti che si contrappongono nei singoli casi, della complessità della vita sociale, della diversità delle circostanze e dell'indeterminatezza degli scopi delle leggi.

Le esigenze di cui al punto (a) potranno trovare soddisfazione nella deduzione logica delle conseguenze di un modello logico del diritto (la base assiomatica del sistema).

Le esigenze di cui al punto (b) dovranno tradursi nella critica e nell'eventuale modificazione di quel modello logico, le quali presuppongono un ragionamento irriducibile ad una deduzione dalla base assiomatica preesistente. Sarà insufficiente il criterio di controllo della validità logica, e si dovranno adottare invece i canoni dell'argomentazione giuridica, ma con il risultato di perfezionare il sistema assiomatico, di predisporre una soluzione deduttiva soddisfacente non solo per il caso considerato ma anche per casi analoghi.

Le possibilità di conciliare queste esigenze dipenderà dalla concreta possibilità di un continuo adattamento e aggiornamento del sistema, possibilità che si collega da un lato alle caratteristiche tecniche del sistema (la trasparenza, la facilità di modificare la base di conoscenza ecc.), d'altro lato al suo contesto di utilizzo.

## **2. Scienze naturali e scienze giuridiche**

Vi è un'importante differenza tra la conoscenza da trasmettere ad un sistema informatico-giuridico "intelligente" e quella da trasmettere ad un sistema destinato ad operare nel settore delle scienze empiriche.

Le conoscenze da trasmettere ad un sistema esperto in una scienza empirica (ad esempio, la medicina, la geologia,

la chimica ecc.) sono fondamentalmente teorie scientifiche, che esprimono una rappresentazione o un modello predittivo del settore della realtà oggetto della scienza di cui si tratta<sup>12</sup>.

Invece, la conoscenza da trasmettere ad un sistema esperto giuridico non può essere che la scienza giuridica (o, se non si vuole usare la parola "scienza" la dottrina o la dogmatica giuridica), il sapere che viene trasmesso a coloro che acquistano una formazione giuridica e che viene usato per affrontare i problemi giuridici.

La scienza giuridica non solo ha per oggetto un fenomeno sociale (e quindi presenta i problemi metodologici propri delle scienze sociali), ma è caratterizzata da un peculiare statuto epistemologico<sup>13</sup>. Infatti, anche se si possono trovare talune analogie tra i metodi delle scienze empiriche o delle scienze logico-matematiche e i metodi delle scienze giuridiche (alle quali gli studiosi che si sono occupati del problema hanno attribuito un diverso peso), la scienza giuridica non è riconducibile né al modello delle scienze empiriche né al quello delle scienze logico-matematiche.

---

<sup>12</sup>Come ho già avvertito nelle pagine precedenti, qui uso il concetto di teoria scientifica in un senso molto ampio. Vi comprendo non solo rappresentazioni complete, coerenti, matematizzate, di una certa classe di fenomeni, ma ogni modello predittivo, e quindi anche le regole di esperienza sulle quali si fa affidamento, in mancanza di una soddisfacente spiegazione scientifica, anche nelle scienze naturali (si pensi, ad esempio, alla medicina). Infatti, in molti contesti bisogna comunque trovare una risposta a domande pressanti, senza attendere lo sviluppo di una teoria soddisfacente.

I sistemi basati sulla conoscenza sono in grado di gestire in modo flessibile grandi quantità di informazioni. Quindi, essi consentono l'uso di sistemi automatici non solo quando si disponga di una teoria scientifica elegante e succinta (in questi casi, spesso, sono sufficienti strumenti informatici tradizionali), ma anche in contesti nei quali si disponga solo di numerose regole d'esperienza, prolisse e casistiche. Ciò comporta, tuttavia, notevoli rischi, e in particolare la possibilità che la base di conoscenza venga a comprendere un ammasso di "conoscenze" non coordinate, difficili da controllare e valutare nelle loro interazioni. In questo caso diventerebbe difficile anche controllare il sistema, prevederne il comportamento, che risulta proprio dall'uso di quelle informazioni, nelle connessioni che si stabiliscono nel processo inferenziale (cfr. WEIZENBAUM J., *Computer Power and Human Reason. From Judgement to Calculation*, cit., specialmente pp. 228 ss.).

<sup>13</sup>Non sarà affrontato in questa sede, neppure superficialmente, il secolare problema della metodologia e dello statuto epistemologico della scienza giuridica. Ci si limiterà a qualche breve accenno ad aspetti significativi per i problemi qui considerati.



Tuttavia, la scienza giuridica si basa su una tradizione bimillenaria e svolge una funzione sociale insostituibile. Pertanto, chi intenda qualificare come "scientifico", il lavoro del giurista si trova, pertanto, in una imbarazzante alternativa:

-Se cerca di estendere anche alle scienze giuridiche i metodi delle scienze logico-matematiche<sup>14</sup> o delle scienze empiriche<sup>15</sup>, o una combinazione dei metodi che caratterizzano i due tipi di scienze<sup>16</sup>, finisce per proporre una radicale revisione dei metodi della scienza giuridica, revisione che la renderebbe incapace di svolgere la propria funzione sociale. Nella migliore delle ipotesi, propone un modello al quale sono riconducibili solo alcuni aspetti del lavoro del giurista.

-Se cerca di raggiungere una ricostruzione metodologica che conservi le peculiarità della scienza giuridica, è costretto ad allontanarsi sia dalle scienze empiriche che dalle scienze logico-matematiche<sup>17</sup>.

Una risposta al problema della specificità della conoscenza giuridica può trovarsi nella considerazione che la pratica e, ad un diverso livello, la scienza del diritto svolgono una funzione etico-politica, seppure in forme specifiche. Esse conducono a - o prefigurano - decisioni di

---

<sup>14</sup>Cfr., ad esempio, ALCHOURRON C., BULYGIN E., *Normative Systems*, Springer Verlag, Wien, New York, 1971, pp. 67 ss., i quali distinguono nell'attività della scienza giuridica due aspetti (a) il problema empirico dell'identificazione degli enunciati che compongono la base assiomatica del sistema giuridico e (b) il problema logico della determinazione delle conseguenze del sistema e della riformulazione della sua base assiomatica.

Una volta identificato l'insieme degli enunciati che compongono la base assiomatica del sistema giuridico, il giurista dovrebbe compiere operazioni di tipo logico: dedurre le conseguenze logiche di quegli enunciati e, eventualmente, proporre una riformulazione della base assiomatica del sistema giuridico, cioè la sostituzione della base assiomatica identificata empiricamente con una base assiomatica più semplice, ma logicamente equivalente all'originaria (in modo che il sistema giuridico, cioè l'insieme degli enunciati deducibili rimanga lo stesso).

<sup>15</sup>Come ad esempio i realisti americani e scandinavi (cfr., per tutti, ROSS A., *Diritto e giustizia*, cit., pp. 29 ss.).

<sup>16</sup>Cfr. BOBBIO N., *Scienza del diritto e analisi del linguaggio*, in "Rivista trimestrale di diritto e procedura civile", 1950, pp. 342-367

<sup>17</sup>Per una considerazione dei rapporti tra scienza giuridica e scienze empiriche, cfr. PATTARO E., *Introduzione al corso di filosofia del diritto*, cit., pp. 314 ss.

questioni pratiche, soluzioni determinate di conflitti di interessi, esprimono scelte che possiamo chiamare, in un senso ampio, politiche. Pertanto presuppongono non solo le conoscenze necessarie per effettuare le operazioni appena menzionate ma anche scelte valutative.

Piuttosto che all'attività scientifica, il ragionamento giuridico può essere ravvicinato al ragionamento etico, può configurarsi come un tipo particolare di argomentazione morale<sup>18</sup>.

In questa prospettiva è rilevante la distinzione tra uso decisionale e uso predittivo dei sistemi informatico-giuridici<sup>19</sup>:

(a) Un sistema destinato ad un uso decisionale deve suggerire una soluzione determinata ad ogni caso concreto. Esso, pertanto, deve basarsi su una ricostruzione determinata del diritto, su un'unica scienza del diritto<sup>20</sup>. Esso presuppone, incorpora, le scelte politico-giuridiche sulle quali quella ricostruzione del diritto si fonda<sup>21</sup>;

(b) Il sistema destinato ad un uso predittivo dovrebbe, in linea di principio, comprendere nella sua base di conoscenza le diverse ricostruzioni del diritto che si contrappongono nella società. La teoria sulla quale si basa il sistema più che una scienza del diritto dovrebbe essere una sorta di sociologia del diritto<sup>22</sup>, per la quale le diverse ricostruzioni del diritto si configurino come elementi di un più complesso fenomeno.

Sono possibili anche soluzioni di compromesso tra le due ipotesi appena menzionate. Ad esempio, un sistema basato sulla conoscenza giuridica potrebbe comprendere non un'unica ricostruzione del diritto, ma le sole ricostruzioni che

---

<sup>18</sup>Tra gli autori che si pongono in questa direzione seppure da diverse prospettive, cfr., tra tutti, SCARPELLI U., *Cosa è il positivismo giuridico*, Edizioni di comunità, Milano, 1968; AARNIO A., ALEXI R., PECZENIK A., *Grundlagen der juristischen Argumentation*, in KRAVIETZ W., ALEXI R., *Metatheorie juristischer Argumentation*, Duncker & Humblot, Berlin, 1983, pp. 9-87.

<sup>19</sup>Cfr. cap. 4 par. 4.2.1.

<sup>20</sup>Non si tratterà necessariamente della posizione assunta da uno scienziato del diritto in senso stretto, cioè da una persona dotata di determinate qualificazioni scientifiche e/o accademiche. Potrà trattarsi della personale ricostruzione operata da chi sviluppa il sistema.

<sup>21</sup>Nel linguaggio di Hart, diremo che un sistema di questo tipo riporta un punto di vista interno rispetto al diritto.

<sup>22</sup>Sulla base di una teoria realistica del diritto.

sembrino migliori, o accettabili, o non aberranti, all'autore del sistema<sup>23</sup>.

### 3. Ragionamento giuridico e metodo assiomatico

Una concezione del ragionamento giuridico come ragionamento morale non preclude la possibilità di un'assiomatizzazione del diritto, e quindi, di creare sistemi basati sulla conoscenza giuridica.

Gli studiosi del diritto e della morale di ispirazione razionalistica concordano nel ritenere che una decisione etica o giuridica sia accettabile solo alla condizione (necessaria ma non sufficiente) di (poter) essere giustificata in un'argomentazione deduttiva. Tuttavia, le premesse ultime di un'argomentazione etica o giuridica non possono essere fondate deduttivamente all'interno dell'argomentazione che muove da esse<sup>24</sup>, né possono essere tutte enunciati suscettibili di verifica empirica<sup>25</sup>.

Il problema della possibilità di giudicare in base a criteri di razionalità le scelte di valore e, in particolare, le scelte che intervengono nel ragionamento giuridico, è da sempre al centro del dibattito filosofico. Senza approfondire questa complessa problematica, si può osservare che, secondo alcune teorie dell'etica e del diritto, le scelte di valore possono essere sottoposte a criteri di razionalità. Tale razionalità si configura diversamente a seconda della particolare concezione di cui si tratta (razionalità come capacità di intuire le verità morali o di coglierle nell'ordine insito nella natura delle

---

<sup>23</sup>In questo modo si attuerebbe un progressivo slittamento da un sistema decisionale ad un sistema predittivo.

<sup>24</sup>In ogni sistema assiomatico (che non consenta di dedurre solo proposizioni logicamente vere), possiamo identificare un insieme di proposizioni primitive, non deducibili da altre proposizioni del sistema.

<sup>25</sup>Come è noto, la c.d. legge di Hume, impedisce di dedurre enunciati prescrittivi da premesse composte esclusivamente di enunciati descrittivi. Inoltre, non è possibile ridurre i concetti prescrittivi a concetti descrittivi mediante definizioni. Così facendo, si incorrerebbe nella c.d. fallacia naturalistica, che vizia i tentativi di definire i concetti normativi (come "buono" o "dovuto") con espressioni descrittive (cfr. MOORE G.E., *Principia Ethica*, Cambridge, 1903 (trad. it. *Principia Ethica*, Bompiani, Milano, 1964)).

cose<sup>26</sup>, come persuasività<sup>27</sup>, come giustificabilità in un contesto discorsivo ideale<sup>28</sup> ecc.), ma non può essere ridotta alla razionalità attribuibile a sistemi informatici, cioè alla capacità di applicare regole di inferenza sintatticamente definite. Tuttavia, la razionalità delle scelte di valore può includere la correttezza logica della giustificazione di tali scelte.

Si è rilevato che il paradigma del sistema deduttivo, che ispira l'intelligenza artificiale, è stato tratto dal contesto della giustificazione. Pertanto, si cercherà esaminare quale tipo di razionalità ci si possa attendere da un sistema informatico giuridico e come questa razionalità si possa ricollegare al problema della giustificazione della decisione giuridica.

---

<sup>26</sup>Come nei pensatori di ispirazione giusnaturalistica.

<sup>27</sup>Come nelle correnti di pensiero che hanno ripreso nel nostro secolo la tradizione della retorica, come la c.d. nuova retorica (cfr. PERELMAN Ch., OLBRECHTS-TYTECA L., *La nouvelle rethorique. Traité de l'argumentation*, cit.) o la c.d. topica (cfr. VIEHWEG T., *Topik und Jurisprudenz. Ein Beitrag zur rechtswissenschaftlichen Grundlagenforschung*, cit.). Perelman, lo studioso che forse ha dato il maggiore contributo allo studio delle argomentazioni non deduttive, fa dipendere la razionalità dell'argomentazione filosofica non dalla persuasività che essa eserciti di fatto su un uditorio determinato, ma dalla sua capacità di convincere un uditorio universale, composto di tutti gli uomini competenti e razionali (cfr. PERELMAN Ch., OLBRECHTS-TYTECA L., *La nouvelle rethorique. Traité de l'argumentation*, cit., p. 40 ss.). Tuttavia, Perelman sembra affermare che solo il filosofo dovrebbe ispirarsi all'uditorio universale, mentre il giudice e il legislatore dovrebbero prendere in considerazione solo i desideri e le convinzioni del popolo che rappresentano (per una critica di questa distinzione, cfr. ALEXI R., *Theorie der juristische Argumentation*, cit. 203 s.).

<sup>28</sup>Si vedano i numerosi lavori di J. Habermas sull'argomento, e, in particolare, HABERMAS J., *Wahrheitstheorien*, in FAHRENBACH H. (a cura di) *Wirklichkeit un Reflexion. Festschrift für W. Schulz*, Pfullingen, 1973, pp. 211-265; J. HABERMAS J., *Theorie des kommunikativen Handelns*, Suhrkamp, Frankfurt, 1981. Tra i filosofi e i teorici del diritto, cfr. ALEXI R., *Theorie der juristische Argumentation*, cit., AARNIO A., ALEXI R., PECZENIK A., *Grundlagen der juristischen Argumentation*, in KRAVIETZ W., ALEXI R., *Metatheorie juristischer Argumentation*, cit., pp. 9-87.

### 3.1. Giustificazione interna e giustificazione esterna della decisione giuridica

La distinzione tra una giustificazione interna ed una giustificazione esterna delle decisioni giuridiche presenta un particolare interesse ai nostri fini. Come afferma J. Wroblewski, "una decisione è giustificata internamente se è dedotta dalle premesse accettate dal decisore, in base alle regole di inferenza che egli considera come valide", "una decisione è giustificata esternamente se le premesse e le regole applicate coerentemente sono corrette in base agli standard accettati dai critici"<sup>29</sup>.

Le conclusioni alle quali giunga un sistema informatico-giuridico, applicando regole di inferenza logicamente corrette, sono giustificate internamente (in quanto dedotte dalla base di conoscenza mediante regole di inferenza logicamente corrette). Esse sono giustificate esternamente in quanto sia giustificata la formalizzazione della base di conoscenza.

Anche a prescindere dal problema del rapporto tra diritto e morale, l'esigenza che le decisioni giuridiche possano essere giustificate (o, come si dice più frequentemente nella nostra cultura giuridica, motivate) è un fondamentale requisito, almeno in ordinamenti giuridici come il nostro<sup>30</sup>.

### 3.2. Modelli di razionalità e di giustificazione nei sistemi informatico-giuridici

Recentemente E. Pattaro<sup>31</sup> riprendendo il pensiero aristotelico ha proposto di distinguere tra una ragione forte (scientifica) e una ragione debole (prudenziale), e più esattamente tra:

---

<sup>29</sup>WROBLEWSKI J., *Justification of Legal Decisions*, in WROBLEWSKI J., *Meaning and Truth in Judicial Decision*, A-THIETO Oy, Helsinki, 1983.

<sup>30</sup>I codici di procedura civile e penale condizionano la validità della sentenza alla sua motivazione. Cfr. art. 361 c.p.c. che stabilisce che le sentenze possono essere impugnate in Cassazione "per omessa, insufficiente o contraddittoria motivazione rispetto ad un punto della controversia, prospettata dalle parti o rilevabile d'ufficio".

<sup>31</sup>PATTARO E., *Introduzione al corso di filosofia del diritto*. Volume secondo, cit., 498 ss. PATTARO E., *Models of Reason, Types of Principles and Reasoning. Historical Comments and Theoretical Outlines*, in "Ratio juris", 2, 1988, pp. 109-122.

(a) "una ragione forte, cioè oggettiva, nel conoscere i principi, i quali saranno pertanto, veri ed evidenti alla ragione";

(b) "una ragione forte, cioè oggettiva, nell'applicare i principi, che deriva conclusioni deduttivamente, che saranno, pertanto, necessarie rispetto ai principi dati";

(c) "una ragione debole, cioè opinabile, nel conoscere i principi, i quali saranno pertanto, verisimili, probabili, semplicemente "accettabili";

(d) "una ragione debole, cioè opinabile, nell'applicare i principi, che deriva conclusioni in maniera non strettamente deduttiva; conclusioni che saranno, pertanto, meramente plausibili (ma non necessarie) rispetto ai principi"<sup>32</sup>.

La razionalità attribuibile ad un sistema automatico non può essere una ragione forte nel conoscere le premesse<sup>33</sup>. Questo concetto di ragione incontra crescenti difficoltà sia nell'epistemologia, sia nella teoria dell'etica contemporanee, che inclinano verso posizioni relativistiche. In ogni caso, non si tratta di una razionalità attribuibile ad un sistema automatico. Un sistema informatico, almeno nel campo giuridico-morale, potrebbe dirsi dotato di questo tipo di razionalità solo nel senso che gli assiomi necessariamente veri facciano parte

---

<sup>32</sup>PATTARO E., *Introduzione al corso di filosofia del diritto. Volume secondo*, cit., 498 ss. Pattaro sovrappone alla distinzione tra scoperta delle premesse e derivazione delle conclusioni, la diversa distinzione tra contesto della scoperta e contesto della giustificazione. Il contesto della scoperta riguarderebbe la scoperta delle premesse, il contesto della giustificazione la derivazione delle conclusioni. A mio parere questa sovrapposizione non è accettabile in quanto anche una conseguenza può essere "scoperta". Le scoperte logico-matematiche consistono non solo nella creazione di nuovi sistemi deduttivi ma anche, nella derivazione di nuovi teoremi, cioè nella scoperta (la deduzione) di conclusioni deducibili in un sistema assiomatico dato, e nella costruzione della loro dimostrazione. La derivazione di nuovi teoremi (non banali) è una scoperta, richiede creatività ed intuizione, in quanto, solo in questo modo si può superare il problema dell'esplosione combinatoria (cfr. cap. 1 par. 3.4.1). Nel campo del diritto e della morale spesso la "scoperta" di una conclusione giuridica o morale precede la giustificazione della stessa nel quadro di un contesto deduttivo, soprattutto quando la giustificazione di una conclusione determinata richieda poi la "scoperta" anche di premesse idonee a fondarla deduttivamente.

<sup>33</sup>Si parlerà di premesse o assiomi, anziché di principi, per evitare confusioni con il concetto di principio nel senso di criterio d'azione che non vincola rigidamente, concetto che sarà ripreso nel seguito (par. 3.1).

della sua base di conoscenza non nel senso che il sistema li abbia "scoperti".

Invece, una ragione debole rispetto alla scoperta delle premesse, può appartenere anche ad un sistema automatico, seppure in una misura molto limitata, in campi definiti e ristretti. Sono stati sviluppati taluni sistemi in grado di compiere operazioni induttive, generalizzazioni, risalendo da casi particolari alla teoria generale in grado di spiegarli (anche se i risultati sinora ottenuti sono largamente insoddisfacenti). Un sistema informatico-giuridico dotato di una razionalità di questo tipo potrebbe ad esempio, estrarre induttivamente le regole alla base di una certa casistica, o ipotizzare, grazie ad un'argomentazione analogica, norme più generali di quelle comprese nella base di conoscenza, in base alle quali disciplinare casi simili a quelli disciplinati da queste ultime norme.

La razionalità forte nell'applicare le premesse, la razionalità deduttiva, come abbiamo visto, costituisce il paradigma cui si ispira prevalentemente l'intelligenza artificiale<sup>34</sup>. Anche i sistemi informatico-giuridici "intelligenti" sin qui sviluppati si limitano, di regola, a questo tipo di ragionamento<sup>35</sup>.

Anche una razionalità debole nell'applicare le premesse può essere attribuita, seppure in misura limitata, ad un sistema informatico-giuridico<sup>36</sup>. Tuttavia, tale razionalità dovrebbe essere ammissibile solo nel contesto della scoperta e non in quello della giustificazione. A prescindere dal modo nel quale il sistema è giunto ad assumere certe premesse e ne ha desunto una determinata conclusione, l'utente dovrebbe sempre poter esigere una giustificazione deduttiva delle conclusioni del sistema<sup>37</sup>.

L'esigenza di usare una razionalità forte, deduttiva, nella giustificazione dell'applicazione delle premesse si

---

<sup>34</sup>Cfr. cap. 1 par. 3.3.

<sup>35</sup>Alcuni dei sistemi informatici giuridici presentati nel cap. 4 cercano di simulare anche il ragionamento analogico. E' particolarmente significativo il tentativo di McCarty (cfr. cap. 4 par. 2.1)

<sup>36</sup>Cfr. cap. 4. par. 5.

<sup>37</sup>Come abbiamo visto, l'esigenza di usare tecniche euristiche superare il problema dell'esplosione combinatoria non si presenta nell'individuazione della disciplina giuridica di un caso determinato (cfr. cap. 2 par. 3.1). Problemi di questo tipo potrebbero presentarsi invece nei sistemi di pianificazione giuridica o anche nei sistemi di consultazione che si proponessero di aiutare l'utente nella ricostruzione del caso.

pone anche (e forse soprattutto) per chi adotti un concetto di razionalità debole rispetto alle premesse<sup>38</sup>.

Sono state preposte due ricostruzioni alternative dei ragionamenti deboli razionali:

(a) *ragionamenti deboli solo rispetto alle premesse*. Ogni conclusione razionale deve poter essere giustificata in un ragionamento deduttivo. I ragionamenti nei quali non compaiono premesse sufficienti a giustificare la conclusione sono razionali (giustificabili) solo se possono essere considerati come entimemi<sup>39</sup> e quindi completati rendendoli deduttivi<sup>40</sup>.

(b) *ragionamenti deboli (anche) rispetto all'applicazione delle premesse*. La giustificabilità deduttiva non è caratteristica necessaria delle argomentazioni razionali. E' inutile, o anzi ingannevole, completare con premesse aggiuntive i ragionamenti deboli, così da trasformarli in ragionamenti deduttivi, cioè in ragionamenti forti per quanto attiene ai metodi di inferenza<sup>41</sup>.

---

<sup>38</sup> Se la deduzione deve essere logicamente corretta, allora tutte le scelte si pongono al livello della individuazione delle premesse. Se le premesse sono solo opinabili, è importante individuarle con precisione in modo che non siano sottratte ad una puntuale valutazione critica.

<sup>39</sup> Come argomentazioni basate su premesse implicite. Gli entimemi solo apparentemente rappresentano argomentazioni non deduttive. Infatti, se le premesse implicite fossero esplicitate, l'argomentazione diverrebbe deduttivamente valida.

<sup>40</sup> Cfr. ad esempio, PECZENIK A., *Legal reasoning as a special case of moral reasoning*, cit., p. 130, il quale illustra questa prospettiva muovendo dal concetto di salto. "Un salto da un insieme di premesse *S* ad una conclusione *q* esiste se e solo se (1) *q* non segue deduttivamente da *S*; e (2) non si può espandere o mutare *S* in modo da ottenere un insieme di premesse *S1* che soddisfi le seguenti condizioni: (a) la conclusione *q* segue deduttivamente da *S1* e (b) *S1* consiste esclusivamente di premesse determinate, premesse presupposte nella cultura in considerazione e premesse provate. Un salto dall'insieme di premesse *S* alla conclusione *q* è ragionevole se e solo se è possibile convertire il salto in un'inferenza deduttiva aggiungendo premesse ragionevoli. Le premesse ragionevoli, convertendo un salto morale in un'inferenza logicamente corretta sono spesso giustificabili mediante la ponderazione e il bilanciamento di vari principi morali".

<sup>41</sup> Cfr. PERELMAN Ch., *Logique juridique. Nouvelle rhétorique*, cit., p. 5, il quale sviluppa la seguente considerazione sulla trasformazione di un'argomentazione in ragionamento deduttivo: "Che cosa abbiamo guadagnato nel trasformare in un sillogismo, che può condurre ad una contraddizione, un'argomentazione non necessaria, ma che permette di



A mio parere è preferibile la prima soluzione, almeno per quanto attiene al ragionamento giuridico. La possibilità di ricondurre una conclusione determinata ad un insieme di premesse sufficienti alla sua deduzione consente di sottoporre la conclusione ad una valutazione puntuale, pone in risalto l'esigenza dell'universalizzabilità del giudizio etico, i valori dell'uguaglianza e della coerenza<sup>42</sup>.

L'esigenza che le decisioni giuridiche possano essere giustificate deduttivamente a partire da (interpretazioni di) norme valide è un fondamentale requisito del ragionamento giuridico<sup>43</sup>. Anche se tutto il diritto non è né sarà mai traducibile in un insieme di assiomi formali, l'esigenza che sia assiomatizzabile il contesto giuridico di ogni decisione determinata è da sempre alla base del lavoro del giurista.

### **3.3. Relatività delle premesse prescrittive e controllo dei sistemi basati sulla conoscenza giuridica**

Tra le premesse del ragionamento giuridico, vi sono non solo proposizioni suscettibili, almeno in linea di principio, di verifica empirica<sup>44</sup>, ma anche norme, contenuti prescrittivi. Il giurista perviene alla ricostruzione delle premesse prescrittive con procedimenti interpretativi, nei quali aspetti valutativi giocano un ruolo decisivo. Si tratta di valori talvolta incompatibili, che debbono essere ponderati, alla luce del particolare problema da risolvere, per ottenere premesse dalle quali sia deducibile una soluzione determinata, senza che sia sempre possibile stabilire un gerarchia definitiva tra di essi. Pertanto, spesso le prescrizioni etiche o giuridiche sono espresse come principi, criteri d'azione che non richiedono di essere seguiti incondizionatamente, ma di essere valutati in

---

giustificare un'opinione plausibile mediante buone ragioni, se non la soddisfazione abbastanza puerile di mostrare che è possibile ricondurre allo stesso schema sillogistico qualsiasi argomento?".

<sup>42</sup>Le premesse della conclusione di cui si tratta saranno tali da giustificare la stessa soluzione anche in altri casi dello stesso tipo.

<sup>43</sup>Si tratta della problematica del c.d. sillogismo giudiziale.

<sup>44</sup>Un'argomentazione deduttiva che conduca a conclusioni prescrittive non può partire solo da premesse descrittive, come si è già osservato. Inoltre, nel diritto, anche le premesse di fatto (apparentemente descrittive), includono spesso elementi valutativi (cfr. WROBLEWSKI J., *Facts in Law, in Meaning and Truth in Judicial Decision*, cit., pp. 104-126).

rapporto con criteri confliggenti per giungere ad una decisione<sup>45</sup>.

La relatività dei valori sui quali si basano le scelte etico-politiche, e la connessa relatività dei compromessi che di volta in volta si raggiungono tra principi confliggenti, si riflette nella relatività e nei limiti di ogni formalizzazione della conoscenza giuridica:

(a) Una rappresentazione formale del diritto, anche quando consenta di dedurre conclusioni determinate (norme applicabili incondizionatamente ai casi sottoposti al sistema automatico), deve essere suscettibile di continue revisioni e essere oggetto di critica nella stessa misura nella quale sono rivedibili e criticabili le premesse (empiriche e normative) di un qualsiasi sillogismo giuridico. Essa comporta valutazioni e quindi responsabilità soggettive.

(b) Quando l'esigenza di valutare principi confliggenti non si possa soddisfare formulando in astratto regole sufficientemente precise, si può pensare di predisporre strumenti che consentano all'utente di effettuare egli stesso una valutazione razionale delle diverse esigenze implicate nel caso<sup>46</sup>,

(c) Un controllo continuo ed approfondito è necessario nei confronti dei sistemi automatici che non si assuma solamente il compito di dedurre automaticamente le conclusioni di premesse date, ma possano anche contribuire allo sviluppo di un modello deduttivo di cui ancora non dispone<sup>47</sup>. In considerazione degli elementi valutativi che sono implicati, ogni conclusione non deduttiva deve essere soggetta ad un controllo da parte di un responsabile. Questo controllo non deve sembrare eccessivamente gravoso. Infatti, in considerazione del principio giuridico di eguaglianza (e

---

<sup>45</sup>Il più recente dibattito filosofico-giuridico ha dedicato notevole attenzione all'opera di Dworkin, il quale ha enfatizzato la distinzione tra regole (*rules*) e principi (*principles*), nel senso appena indicato (cfr. DWORKIN R.M., *Taking Rights Seriously*, Duckworth, London, 1977). Per una ricostruzione della problematica cfr. PATTARO E., *Introduzione al corso di filosofia del diritto. Volume secondo*, cit., p. 419 ss.

<sup>46</sup>Cfr., ad esempio, l'interessante sistema presentato in NAGEL STUART S., *Microcomputers and Judicial Predictions*, in MARTINO A.A., SOCCI NATALI F., *Automated Analysis of Legal Texts*, cit., pp. 681-702.

<sup>47</sup>Qualora la rappresentazione formale del diritto che costituisce la base assiomatica del sistema non comprenda premesse sufficienti a risolvere il caso. Sull'uso di sistemi esperti giuridici nella costruzione di un modello deduttivo, cfr. FIEDLER H., *Expert Systems as a Tool for Drafting Legal Decisions*, in MARTINO A.A., SOCCI NATALI F. (a cura di) *Automated Analysis of Legal Texts*, cit., pp. 607-612.

del principio etico di universalizzabilità), possiamo pensare ai casi che non siano risolvibili mediante un'inferenza deduttiva, come ad occasioni nella quali si perviene ad un'integrazione della base di conoscenza (eventualmente suggerita dal sistema stesso) sufficiente a risolvere non solo il caso problematico, ma anche una classe di casi simili<sup>48</sup>. Le ipotesi nelle quali il sistema non sia in grado di risolvere un caso determinato applicando deduttivamente le conoscenze di cui dispone, risulterebbero quindi anche ipotesi di sviluppo del sistema, di apprendimento (anche se automatizzabile in misura molto limitata, e sottoposto ad un attento controllo).

### **3.4. Conclusione. Ragionamento giuridico e razionalità dei sistemi informatico-giuridici intelligenti**

I sistemi informatico-giuridici intelligenti dispongono fondamentalmente di una razionalità forte nel derivare conclusioni dalle premesse. Il solo limite a riguardo è rappresentato dalla esplosione combinatoria.

In misura molto più limitata possiamo attribuire ad alcuni sistemi anche una razionalità debole nello scoprire premesse (ragionamento per esempi, analogia) o nel derivare conclusioni (metodi euristici). Si è già rilevato che il problema dell'esplosione combinatoria non è così pressante nel diritto, almeno per quanto attiene ai sistemi di analisi giuridica. Per questi sistemi non sembra che si ponga il problema di implementare una razionalità debole nella scoperta delle premesse. Il vero problema è quello della costruzione di un modello deduttivo, della scoperta delle premesse.

Il giurista non compie solo né prevalentemente operazioni deduttive. Tuttavia ogni decisione giuridica deve essere giustificata in un contesto deduttivo. Non si tratterà dell'intero ordinamento giuridico ma delle premesse della decisione di cui si tratta. Lo sviluppo di una base di conoscenza richiede un maggior sforzo di astrazione, nel senso che occorre individuare un contesto deduttivo sufficiente a fondare la soluzione non solamente di un caso determinato ma di tutte le questioni proponibili nel settore giuridico che si intende affidare al sistema (più

---

<sup>48</sup>Abbiamo visto che anche i ragionamenti non deduttivi possono essere resi deduttivi introducendo ulteriori premesse. La giustificazione di un ragionamento non deduttivo può essere ricondotta alla giustificazione esterna di queste premesse aggiuntive. Le premesse aggiuntive possono essere, eventualmente, proposte all'utente dal sistema. Mi sembra assai pericolosa, invece, l'ipotesi nel quale sia il sistema stesso ad assumere autonomamente premesse nuove, senza espressa autorizzazione.

realisticamente, un insieme significativo di quelle questioni). Si tratterà della assiomatizzazione di un sottosistema, di un settore ristretto dell'ordinamento giuridico.

Se questa assiomatizzazione è possibile, allora anche nel diritto è possibile l'uso "improprio" del metodo deduttivo che caratterizza l'intelligenza artificiale, l'uso del metodo deduttivo per scoprire conclusioni "nuove", per rispondere a domande di cui non si dispone ancora della risposta.

L'esame della razionalità attribuibile ad un sistema automatico consente di ribadire la tesi già illustrata in precedenza: il ragionamento di un sistema basato sulla conoscenza giuridica nella soluzione di un problema determinato deve essere fondamentalmente un ragionamento deduttivo. I problemi da risolvere grazie ad una razionalità debole debbono essere affrontati nel quadro dello sviluppo di un modello deduttivo, della modifica della base di conoscenza del sistema. Anche in questa funzione, tuttavia, è possibile un limitato contributo da parte dello stesso sistema automatico.

#### **4. Interpretazione del diritto e rappresentazione della conoscenza**

L'aspetto centrale del lavoro del giurista (sia del giurista pratico sia dello scienziato del diritto), sta nella c.d. *interpretazione*, cioè nella determinazione della disciplina giuridica da applicare ad un caso concreto o ad una classe di casi, determinazione da compiersi sulla base di (a) documenti che riportano atti normativi, ma anche (b) regole, principi, canoni che, anche se non riportati in documenti ufficiali, appartengono alla cultura giuridica e (c) valutazioni dell'interprete. Il diritto è il risultato dell'interpretazione e, quindi, l'interpretazione costituisce la premessa necessaria della rappresentazione formale del diritto in una base di conoscenza.

##### **4.1. Il concetto di interpretazione**

Il concetto di interpretazione, inteso nel senso più ampio, comprende nel insieme di attività che "non si risolvono nel mero attribuire significato ad un documento legislativo o all'insieme dei documenti legislativi, ma includono operazioni complesse: (a) l'individuazione di un segmento del discorso legislativo, mediante scomposizione e ricomposizione dei documenti con riferimento ad una <<sistematica>>, suscettibile di essere interpretato come

esprimente la c.d. <<norma del caso>>; (b) la attribuzione a questo segmento del discorso legislativo di un significato, decidendo il valore da dare ai vocaboli indefiniti contenuti nei suoi enunciati, con riferimento al linguaggio ordinario; (c) la risoluzione di antinomie o conflitti tra leggi, con riferimento a modi di concepire il diritto ed a ideologie relative alla sua coerenza; (d) la integrazione delle legge e la ricerca eventuale di norme regolatrici del caso, con riferimento a modi di concepire il diritto come completo o incompleto, e ad ideologie circa i modi di integrazione delle leggi e la ricerca di diritto extralegislativo.

Inoltre l'area semantica delle locuzioni <<interpretazione del diritto>> e <<interpretazione della legge>> include anche le operazioni di descrizione dei comportamenti e delle situazioni che si intende qualificare giuridicamente; si tratta di descrizioni apparenti e in realtà di pre-qualificazioni: infatti i comportamenti e le situazioni vengono indicati, positivamente o negativamente, mediante vocaboli che compaiono nel discorso legislativo (o nel discorso in cui si formulano le norme extra-legislative risultato di integrazione o ricerca del diritto) onde pre-constituire la qualificazione giuridica che ai comportamenti o alle situazioni si vuole attribuire. La interpretazione della legge e la descrizione (o, come si dice anche, la <<ricostruzione>>) dei c.d. <<fatti>> (comportamenti o situazioni) interagiscono e sono sovente indistricabilmente intrecciate, tanto che si parla di <<interpretazione>>, sovente, per riferirsi ad entrambe"<sup>49</sup>.

Alcuni autori hanno proposto diversi, più ristretti concetti di interpretazione.

Ad esempio, Alchourron e Bulygin<sup>50</sup> adottano un concetto logicizzato della interpretazione giuridica. Questi autori riducono l'interpretazione alla riformulazione della base assiomatica del sistema giuridico e alla derivazione delle conseguenze deducibili nel sistema stesso, operazioni che non modificano i contenuti del sistema. In questo modo, essi dimenticano che il giurista - usando i criteri di riconoscimento di un determinato ordinamento giuridico, e, più in generale, le norme di competenza - può identificare non norme (significati prescrittivi), né enunciati già dotati di significato, ma solo testi ai quali ascrivere un significato. Il primo e fondamentale compito del giurista,

<sup>49</sup>TARELLO G., *L'interpretazione della legge*, cit., 1980, p. 32-33.

<sup>50</sup>ALCHOURRON C., BULYGIN E., *Normative systems*, cit., 1970, pp. 67 ss. che, come abbiamo visto riducono la scienza giuridica all'identificazione di una base assiomatica e alla sua riformulazione e alla deduzione delle conseguenze

una volta che si siano individuati i testi normativi<sup>51</sup>, non è la deduzione delle conseguenze da quei testi<sup>52</sup>, né la riformulazione del sistema, cioè la sostituzione di una nuova, più semplice e razionale base assiomatica alla base assiomatica originaria<sup>53</sup>, ma l'attribuzione di un significato a quei testi<sup>54</sup>.

Altri invece, come in particolare J. Wroblewski, parlano di interpretazione solo per designare le situazioni nelle quali c'è un dubbio sul significato di un testo giuridico e nelle quali, pertanto, il testo non può essere usato con "un significato dato immediatamente"<sup>55</sup>. Neppure questo concetto di interpretazione sembra accettabile ai nostri fini. Infatti, allo stato delle conoscenze linguistiche, non disponiamo di una teoria in base alla quale distinguere i casi in cui l'attribuzione di un significato ad

---

<sup>51</sup>Si può determinare quali siano i testi normativi solo in base all'interpretazione delle norme di competenza. Tuttavia, di regola questo problema non presenta particolari difficoltà (vi sono raccolte ufficiali ecc.).

<sup>52</sup>La determinazione delle conseguenze di un testo in linguaggio naturale, poiché non sono applicabili regole di inferenza definite sintatticamente, presuppone una determinazione del significato. Inoltre, se si vogliono applicare le regole d'inferenza della logica, quel significato deve essere espresso in un linguaggio formale.

<sup>53</sup>Anche questa operazione può farsi, rispetto ad un testo in linguaggio naturale, solo dopo che se ne sia determinato il significato.

<sup>54</sup>Cfr. anche ALCHOURRON C., *Systematisation and Change in the Science of Law*, in ECKHOFF T., FRIEDMAN L.M., UUSITALO J. (a cura di), *Vernunft und Erfahrung in Rechtsdenken der Gegenwart*, "Rechtstheorie", Beiheft 10, Duckner & Humblot, Berlin, 1986, pp. 171-184, che risponde alle osservazioni critiche di AARNIO A., *On Changes on the Systematics of Law*, in ECKHOFF T., FRIEDMAN L.M., UUSITALO J. (a cura di), *Vernunft und Erfahrung in Rechtsdenken der Gegenwart*, cit., pp. 161-170, affermando per "norme" deve intendersi enunciati già dotati di un significato (e quindi non avrebbe senso parlare di interpretazione di norme). E certo, tuttavia, che enunciati già dotati di significato non sono la premessa, ma il risultato del lavoro del giurista, del quale l'interpretazione è l'aspetto fondamentale; che il problema della determinazione delle "norme", nel senso in cui ne parla Alchourron non è un problema empirico; e che tra interpretazione, deduzione di conseguenze logiche di norme interpretate, riformulazione del sistema normative, sono attività, benché analiticamente separabili, difficilmente distinguibili nell'attività del giurista.

<sup>55</sup>Cfr. WROBLEWSKI J., *Justification of Legal Decisions*, cit., p. 73 s.

un'espressione linguistica possa dirsi dubbia<sup>56</sup>. La distinzione tra casi di dubbio e casi di certezza nella rilevazione del significato di un testo è puramente psicologica, quindi inutilizzabile nel contesto dello sviluppo di sistemi informatici.

Pertanto, nella discussione dei problemi informatico-giuridici, sembra preferibile il concetto più ampio di interpretazione introdotto sopra.

#### 4.2. Interpretazione e testi

I sistemi informatici intelligenti non sono in grado di ragionare direttamente in linguaggio naturale e, comunque l'interpretazione giuridica comporta valutazioni che non possono essere lasciate ad un sistema automatico.

Pertanto, nei sistemi basati sulla conoscenza giuridica, i problemi interpretativi (i testi giuridici sono formulati nel linguaggio naturale) debbono essere trattati non nel contesto della soluzione di concreti problemi giuridici (come avviene nell'attività del giurista), ma nel contesto nella assiomatizzazione-formalizzazione della conoscenza giuridica<sup>57</sup>, cioè nella preparazione della base di conoscenza del sistema.

Infatti, tradurre dei testi giuridici in un linguaggio formale significa determinare il significato di quei testi e specificarlo con precisione, significa interpretare i testi giuridici e formulare il risultato dell'interpretazione in un linguaggio formale. La formalizzazione-attività è quindi

---

<sup>56</sup>Si potrebbe forse sostenere che l'interpretazione di un testo è dubbia quando le regole della sintassi e della semantica del linguaggio nel quale il testo è formulato consentano più ricostruzioni alternative del suo significato. Si tratta però di una formulazione vuota: non possediamo una grammatica completa per alcun linguaggio naturale e gli studi sulla semantica dei linguaggi naturali sono solo agli inizi. I frammenti di teorie del linguaggio di cui disponiamo conducono a rilevare ambiguità in qualsiasi enunciato la cui struttura presenti una qualche complessità.

<sup>57</sup>C'è stato chi ha prospettato la possibilità di usare direttamente il linguaggio naturale per la rappresentazione della conoscenza giuridica, in relazione alle prospettive dei elaboratori della quinta generazione (cfr. HEATHER M.A., *Future Generation Computer Systems*, in MARTINO A.A., SOCCI NATALI F. (a cura di) *Automated Analysis of Legal Texts*, cit., pp. 643-660). Questa soluzione, a mio parere, non è né possibile (come abbiamo osservato nel cap. 1 vi sono limiti delle applicazioni dell'intelligenza artificiale insuperabili con le tecnologie oggi disponibili), né auspicabile (tutte le scelte valutative inerenti all'interpretazione giuridica sarebbero devolute al sistema automatico).

una forma particolare di interpretazione-attività mentre la formalizzazione-risultato è una forma particolare di interpretazione-risultato.

L'interpretazione giuridica non è interamente riducibile alla problematica più generale della comprensione e della formalizzazione del linguaggio naturale. Infatti, l'interpretazione giuridica non è volta a ricostruire tutti i significati attribuibili ad un enunciato in base alle regole linguistiche, ma si configura invece come una complessa attività con la quale l'interprete, sulla base dei significati attribuibili al testo, delle intenzioni attribuibili all'emittente, ma anche di considerazioni (almeno in parte valutative) circa il contesto normativo nel quale il testo si colloca, le conseguenze sociali morali, economiche, ecc. di diverse possibili interpretazioni, contribuisce alla produzione del diritto<sup>58</sup>, contribuisce alla definizione delle regole di comportamento di una determinata società.

Taluni studiosi che si sono occupati dei sistemi esperti nel diritto, sono stati portati ad identificare il diritto con gli atti normativi, cioè con i documenti che riportano leggi, sentenze, atti amministrativi<sup>59</sup>. Questa concezione del diritto si unisce talvolta ad un trasferimento non meditato del linguaggio e dei problemi dell'intelligenza artificiale, nell'informatica giuridica. Abbiamo rilevato che il problema fondamentale nello sviluppo di un sistema esperto è quello della "cattura" e della rappresentazione della conoscenza: la conoscenza va "prelevata" dai testi o dall'esperto, e va trasferita nel sistema, che potrà usarla per risolvere i problemi.

---

<sup>58</sup>Cfr. PATTARO E., *Introduzione al corso di filosofia del diritto. Volume secondo*, cit., p. 203 ss.

<sup>59</sup>Negli studi informatico-giuridici il problema dell'interpretazione raramente è stato oggetto di un'adeguata considerazione. Così in SUSSKIND R.E., *Expert Systems in Law*, cit., 1988 p. 36 s., la prima monografia sui sistemi esperti giuridici specificamente intesa a considerare i profili teorico-giuridici, si afferma che la base di conoscenza dei sistemi esperti giuridici dovrebbe consistere fondamentalmente di asseriti del diritto (*law statements*), cioè di enunciati che descrivono formulazioni giuridiche (*law formulations*), cioè "la carta e i caratteri noti come legislazione e repertori di giurisprudenza", "entità linguistiche, formulate linguisticamente come enunciati scritti composti dei parole scritte". In questa formulazione vengono confusi (a) i testi, (b) il significato ad essi ascritto c) la riformulazione formale di quel significato destinata a fungere da base di conoscenza (c) il significato che l'utente del sistema ascrive a quella riformulazione formale (d) il significato che la base di conoscenza riceve dal trattamento automatico cui è sottoposta.



Non stupisce che si sia intravisto nel diritto il campo applicativo ideale per la realizzazione di sistemi esperti. In particolare, nel diritto non si presenterebbe il difficile problema della "cattura" e della rappresentazione della conoscenza: i testi legislativi (integrati eventualmente dai repertori di giurisprudenza) conterrebbero la conoscenza necessaria e sufficiente a realizzare un sistema esperto giuridico. La realizzazione di una base di conoscenza giuridica consisterebbe nella traduzione di quei testi nel linguaggio formale per la rappresentazione la conoscenza. In questa prospettiva, è dubbia l'utilità non solo di un apporto di conoscenze teorico-giuridiche nella formalizzazione del diritto, ma anche del contributo del giurista. L'"esperto" giuridico sarebbe il legislatore o il giudice, che ha già espresso e documentato esaurientemente la conoscenza giuridica, realizzando così i presupposti del lavoro dell'ingegnere della conoscenza<sup>60</sup>. Questo approccio al problema della rappresentazione della conoscenza giuridica dimentica però che:

- non tutto il diritto è contenuto nella legge o nei repertori di giurisprudenza;

- la traduzione di un testo giuridico in una base di conoscenza è il passaggio da un linguaggio naturale ad un linguaggio formale, nel quale debbono essere risolti problemi interpretativi<sup>61</sup>;

- tale traduzione presenta non solo i problemi del trattamento informatico del linguaggio naturale, ma anche e soprattutto gli specifici problemi dell'interpretazione e dell'applicazione del diritto.

La prospettiva semplificatrice che si è illustrata è estremamente pericolosa, in quanto nasconde le scelte e le

---

<sup>60</sup>Cfr. SERGOT M.J., SANDRI F., KOWALSKI R.A., KRIWACZEK F., HAMMOND P., CORY H.T., *British Nationality Act as a Logic Programme*, cit., p. 383: "Il problema della esplicitazione della conoscenza è quasi completamente assente nella formalizzazione della legislazione. Per la sua natura, il diritto è ben documentato; le sue disposizioni sono scritte, e dove non lo sono, decisioni di casi precedenti sono registrate in modo che si possa in futuro far riferimento ad esse". Una diversa, più matura consapevolezza dei problemi e dei limiti delle applicazioni dell'intelligenza artificiale nel diritto, ispira invece lavori successivi provenienti dal gruppo di ricerca dell'Imperial College (cfr. in particolare KOWALSKI R.A., SERGOT M.J., *The Use of Logical Models in Legal Problem Solving*, cit.).

<sup>61</sup>La sintassi del linguaggio formale adottato, obbliga a precisare il significato delle locuzioni che debbono essere sostituite da espressioni formali, e solo ciò che sia stato tradotto in strutture formali è manipolabile (e in questo senso comprensibile) da parte dell'elaboratore.

responsabilità che l'interpretazione, e quindi la rappresentazione formale della conoscenza giuridica, comportano. Essa sembra il frutto di una concezione riduttiva del significato: il significato non è il risultato di atti di interpretazione (nel quadro delle regole che definiscono un contesto linguistico e culturale) ma è una misteriosa sostanza che inerisce alle parole, e che può essere ap проблемaticamente trasferita da un supporto (il testo originario) ad un altro (la base di conoscenza espressa in un linguaggio formale)<sup>62</sup>.

Ricondotta invece alla tematica dell'interpretazione, la preparazione della base di conoscenza di un sistema informatico-giuridico intelligente mostra invece la problematicità, il rilievo politico e giuridico, la responsabilità che ad essa si collegano.

#### **4.3. Interpretazione e competenze dell'ingegnere della conoscenza giuridica**

La formalizzazione del diritto al fine dello sviluppo di un sistema di conoscenza è un tipo di interpretazione, ma presenta specifici problemi.

L'interpretazione operata dal giurista pratico in vista della determinazione della disciplina di un caso particolare, si limita ad una attribuzione di significato al testo legislativo (più in generale, alla ricostruzione della disciplina giuridica) per gli aspetti che rilevano ai fini della soluzione del caso che gli è sottoposto<sup>63</sup>.

L'interpretazione operata dallo scienziato del diritto, seppure riferita ad una pluralità di casi, è limitata all'oggetto della proposta interpretativa formulata dal

---

<sup>62</sup>Cfr. STAMPER R.K., BACKHAUS J., ALTHAUSE K., *Expert Systems-Lawyers Beware*, cit., 1988, p. 317.

<sup>63</sup>Ad esempio, se il problema è determinare la liceità della registrazione automatica delle operazioni compiute dal lavoratore nell'interazione con un determinato sistema di calcolo, ai sensi dell'art. 4 dello statuto dei lavoratori (L. 20 maggio 1970 n. 300), il giudice potrà limitarsi a determinare la portata dell'articolo in questione rispetto a questa ipotesi. Benché l'articolo vieti, in generale, "impianti audiovisivi e altre apparecchiature di controllo a distanza dell'attività dei lavoratori" non sarà necessario considerare il problema della liceità di altre apparecchiature di controllo a distanza, o dell'elaborazione automatica di dati relativi al lavoro svolto, ma introdotti manualmente nel sistema di calcolo.

giurista, al livello d'astrazione proprio di quella proposta<sup>64</sup>.

La costruzione del teorico del diritto considera problemi ancora più astratti, proponendosi di elaborare gli strumenti concettuali del giurista<sup>65</sup>.

Un sistema informatico-giuridico intelligente è chiamato a risolvere una pluralità indeterminata di casi, in tutti i loro aspetti. Le operazioni interpretative non possono essere delegate al sistema che in una misura molto limitata. Pertanto il sistema deve disporre di un'interpretazione-risultato che formulata con tale precisione da consentire una decisione automatica anche di casi futuri (senza ricorrere ad ulteriori operazioni interpretative). Quindi, l'ingegnere della conoscenza giuridica ha il compito gravoso di formulare un'interpretazione che:

- raggiunga la precisione delle interpretazioni riferite a casi singoli (in questo senso la sua attività si avvicina a quella del giurista pratico);

- copra tutto il settore nel quale deve essere usato il sistema informatico (sotto questo aspetto la sua attività si avvicina invece a quella dello scienziato del diritto);

- rappresenti fedelmente il diritto (ciò che il responsabile del sistema intende per diritto), ma sia strutturata in modo da poter essere aggiornata e corretta facilmente e da poter essere usata efficientemente dal motore inferenziale (sotto quest'ultimo profilo l'attività dell'ingegnere della conoscenza giuridica presuppone competenze che si avvicinano a quelle del teorico del diritto).

#### 4.4. Il problema dell'interpretazione del fatto

L'esigenza di raggiungere una rappresentazione del diritto sufficientemente precisa da consentire la soluzione di un numero indeterminato di casi futuri, pone un grave problema per quanto attiene alla c.d. interpretazione del fatto. Come abbiamo visto, l'interpretazione del fatto è solo in parte un'operazione descrittiva: nel qualificare un fatto nei termini di una previsione normativa, da un lato si

---

<sup>64</sup>Ad esempio, lo scienziato del diritto potrà esaminare, in generale, il problema della liceità dell'uso di strumenti di controllo a distanza dell'attività dei lavoratori, o trattare il problema della liceità del controllo mediante strumenti informatici (si tratta di controllo a distanza?), o considerare casi più specifici, commentando, ad esempio, una sentenza in materia.

<sup>65</sup>Il problema del rapporto tra teoria del diritto e sistemi informatico-giuridici intelligenti sarà considerato nel par. 8.

riscontrano certe caratteristiche di quel fatto<sup>66</sup>, dall'altro si completa l'interpretazione della norma, precisandone i contenuti che consentono di sussumere il fatto nella previsione della norma.

Possiamo pensare a diverse soluzioni per questo problema, nessuna pienamente soddisfacente:

(a) Una prima soluzione consiste nel trasferire completamente sull'utente l'onere della "interpretazione del fatto". Il sistema richiede che l'utente qualifichi i fatti rilevanti direttamente nei termini del linguaggio legislativo, o in una riformulazione che si ponga allo stesso livello d'astrazione del testo legislativo. Ad esempio, supponiamo che la nostra base di conoscenza contenga la regola (estratta dall'art. 2043 cc.):

**se "una persona X compie un fatto Y" e,  
 "il fatto Y causa un danno D alla persona Z", e  
 "il danno D è ingiusto", e  
 allora  
 "la persona X deve risarcire il danno D alla persona Y".**

Supponiamo di proporre un quesito relativo alla vendita di un'elaboratore difettoso, che ha causato la distruzione dei programmi dell'acquirenti.

Se il sistema si limita a porre all'utente domande del tipo:

**"il danno (distruzione di archivi informatici) è ingiusto?".**

la stessa interpretazione della norma è trasferita sull'utente. Infatti nella preparazione della base di conoscenza ci è limitati alla rappresentazione formale degli aspetti sintattici rilevanti per l'applicazione del calcolo logico usato<sup>67</sup>, mentre le espressioni non sintattiche, che per il sistema sono sequenze di caratteri arbitrarie, vengono usate per porre domande all'utente<sup>68</sup>. Si osservi che l'utente, nei casi problematici, può effettuare una corretta

---

<sup>66</sup>Talvolta anche la descrizione dei fatti può presentare aspetti valutativi (cfr. WROBLEWSKI J., *Facts in Law*, cit.).

<sup>67</sup>Supponiamo di aver adottato la logica predicativa come linguaggio per la rappresentazione della conoscenza (omettiamo i quantificatori universali).

<sup>68</sup>Il sistema non dispone di alcuna rappresentazione del significato della parola "ingiusto". Si tratta di una sequenza di caratteri priva di significato. E' l'utente, che le attribuisce un significato nel rispondere alla domanda che comprende questa parola.

qualificazione dei fatti rilevanti solo nella consapevolezza della conseguenza giuridica che discende dalla qualificazione. E' opportuno, pertanto, assicurare all'utente la possibilità di essere informato dal sistema in ogni momento sul senso dell'operazione che il sistema sta compiendo (sulla regola che si stà applicando).

(b) Una seconda soluzione consiste nell'integrare la base di conoscenza con definizioni di origine dottrinale, giudiziale, o frutto di una interpretazione compiuta nella creazione della base di conoscenza. Il sistema potrà determinare se un certo fatto debba essere qualificato nei termini del *definiendum*, chiedendo all'utente di qualificarlo nei termini del *definiens*. Ad esempio, il sistema disporrà della definizione in base alla quale un danno è ingiusto se viola un diritto, o un interesse meritevole di tutela ecc. Quindi, per determinare se la distruzione degli archivi di Mario realizzi un danno ingiusto, il sistema potrà chiedere all'utente se questo fatto rappresenti la violazione di un diritto, oppure la lesione di un interesse meritevole di tutela ecc., e dedurre l'ingiustizia se l'utente risponde affermativamente ad una di queste domande. Questa soluzione trasferisce sull'ingegnere della conoscenza l'onere di una definizione dei termini del linguaggio normativo<sup>69</sup>, ma continua a costringere l'utente ad esprimersi nel linguaggio del sistema. Un perfezionamento ulteriore potrebbe ottenersi cercando di dare una soluzione più organica al problema della rappresentazione dei rapporti concettuali, riconducendo i diversi concetti ad una combinazione definita di un numero ristretto di concetti primitivi<sup>70</sup>. Tuttavia, ogni costruzione concettuale di questo tipo si rivela estremamente difficile da realizzare, e si basa inevitabilmente anche su scelte e preferenze individuali<sup>71</sup>.

c) Una terza soluzione, ancor più difficile da realizzare, consiste nell'implementare un trattamento

---

<sup>69</sup>Nell'introdurre queste definizioni l'ingegnere della conoscenza giuridica formula una sorta di definizione dottrinale

<sup>70</sup>Si tratta di un ideale che è stato perseguito anche da taluni indirizzi della scienza giuridica, che hanno cercato di sviluppare dei sistemi dei concetti giuridici, nella traccia della giurisprudenza dei concetti.

<sup>71</sup>Nell'intelligenza artificiale questa prospettiva è fatta propria da teorie come quella della dipendenza concettuale (cfr. SCHANK R.C., *Conceptual Information Processing*, Elsevier, New York, 1975). Le ricerche in questa direzione non hanno ancora dato risultati definitivi.

automatico del linguaggio naturale<sup>72</sup>. All'utente è possibile esprimersi nel suo linguaggio, anche se diverso da quello nel quale i contenuti giuridici sono stati formalizzati. Il sistema è in grado di interpretare le espressioni usate dall'utente per controllare se esse corrispondano a circostanze giuridicamente rilevanti (alla rappresentazione dei contenuti giuridici nella sua base di conoscenza). Possiamo immaginare un'interazione tra uomo e sistema informatico nella quale (a) l'utente affermi: "il programma difettoso mi ha distrutto gli archivi", (b) il sistema comprenda che può trattarsi di un problema di responsabilità civile, e (c) proponga all'utente domande ulteriori così da ottenere informazioni relative ai fatti rilevanti ai fini della realizzazione di una fattispecie di responsabilità civile. La realizzazione integrale di questo tipo di interazione tra sistema e utente presuppone che il sistema informatico sia in grado non solo di trattare il linguaggio naturale, ma anche di effettuare le valutazioni interpretative necessarie per rapportare il linguaggio dell'utente alle strutture rappresentative della base di conoscenza. Questa condizione non sembra completamente realizzabile, per ragioni non solo tecnologiche, ma anche linguistiche e filosofico-giuridiche. Pertanto, all'utente deve essere consentito controllare il rapporto tra il linguaggio dell'utente e la traduzione formale effettuata dal sistema. Se così non fosse, la "comprensione" automatica del linguaggio naturale si tradurrebbe in un aumento dell'imprevedibilità del sistema, anziché in un aumento della facilità d'uso e della trasparenza.

## **5. Conoscenze giuridiche pubbliche e private**

Sia la conoscenza giuridica pubblica, sia la conoscenza privata del giurista esperto dovrebbero essere rappresentate nella base di conoscenza di un sistema informatico-giuridico "intelligente". La conoscenza giuridica pubblica è quella accessibile in testi pubblicamente disponibili: gazzette ufficiali, repertori giurisprudenziali, scritti dottrinali ecc. La conoscenza giuridica privata è quella legata alla concreta esperienza dell'operatore giuridico. Ad esempio, la conoscenza privata di un avvocato potrà comprendere conoscenze concernenti (a) il concreto funzionamento delle istituzioni giudiziarie e amministrative, in particolare nella sua circoscrizione, (b) le strategie da adottare nei

---

<sup>72</sup>Il sistema informatico-giuridico che ha raggiunto i risultati più significativi in questa direzione è stato il progetto LEX, già illustrato nel cap. 4 par. 3.2.

rapporti con i clienti, le controparti, i giudici (c) le tecniche per la redazione degli atti ecc. Le conoscenze private possono essere fornite solo dal giurista esperto, ma il ricorso all'esperto è necessario, al fine della preparazione di una base di conoscenza giuridica, anche per ottenere una soluzione di questioni interpretative. Infatti, come si è già osservato, le leggi, i repertori e i testi dottrinali non contengono tutti gli elementi necessari per giungere ad una rappresentazione del diritto così precisa, priva di vaghezze e ambiguità, da rappresentare una base di conoscenza.

Pertanto, nello sviluppo di un sistema esperto giuridico, si dovranno utilizzare, in ogni caso, tutte le conoscenze pubbliche disponibili (specificando, per ognuna di esse, la fonte). Quando esse non siano sufficienti, si dovrà ricorrere anche alla collaborazione di giuristi esperti, non solo per avere conoscenze legate alla pratica, ma soprattutto per ottenere un'affidabile soluzione alle questioni interpretative non risolte in modo soddisfacente e/o completo nelle fonti pubbliche.

Si è recentemente affermato che la distinzione tra conoscenza giuridica pubblica e privata dovrebbe essere sostituita da quella tra

- conoscenza accademica, che può essere acquisita da chiunque abbia una formazione giuridica, in base allo studio delle fonti del diritto e dei relativi commentari, e

- conoscenza pratica (*experiential*), che, anche se talvolta resa disponibile in testi giuridici, manuali pratici, o materiali interni, viene acquisita fondamentalmente nella pratica quotidiana del lavoro giuridico. I sistemi esperti giuridici potrebbero comprendere solo o principalmente la conoscenza accademica<sup>73</sup>.

Questa proposta non sembra accettabile. Infatti, non si vede perché nella realizzazione un sistema esperto giuridico non sia possibile avvalersi anche di conoscenze relative alla pratica, soprattutto se pubblicamente accessibili (ad esempio, i modelli e i formulari per la redazione di atti giuridici rappresentano un ausilio insostituibile per lo sviluppo di un sistema per la redazione automatica di quegli atti). La stessa distinzione tra conoscenza accademica e pratica non è molto chiara: è accademico solo ciò che è insegnato nelle facoltà di giurisprudenza, o anche ciò che è insegnato nei corsi postuniversitari che preparano all'esercizio delle professioni giuridiche o in corsi d'aggiornamento per le professioni giuridiche?.

---

<sup>73</sup>SUSSKIND R.E., *Expert systems in Law*, cit., p. 46 ss.

## 6. Conoscenze e posizioni giuridiche

La conoscenza memorizzata nella base di conoscenza di un sistema giuridico può essere chiamata "privata" in un diverso senso.

Si è distinto tra sistemi informatico giuridici con finalità decisionale e sistemi con finalità previsionale<sup>74</sup>.

Il responsabile del sistema con finalità decisionale si aspetta che questo fornisca non tutte le ricostruzioni del diritto possibili (o quelle sostenute nella dottrina o nella giurisprudenza, o dalla maggior parte dell'una o dell'altra), ma la soluzione "migliore" (cioè quella fondata sulla ricostruzione del diritto che egli stesso accoglie). Quindi, questi sistemi contengono una conoscenza "privata" in quanto basata su determinate scelte politico-giuridiche.

L'uso dell'espressione "conoscenza" per indicare il contenuto di questi sistemi sembra presupporre una forma di cognitivismo etico, cioè la tesi che i valori, anziché espressioni di scelte personali, una realtà conoscibile e descrivibile. Sarebbe forse più opportuno parlare, anziché di base di "conoscenza", di base di "posizioni", di "credenze" o di "dottrine giuridiche", in modo da evitare che lo sviluppo di sistemi informatici sembri collegato ad una determinata teoria dell'etica e della scienza giuridica.

## 7. Conoscenze profonde e superficiali

McCarty ha introdotto nel dibattito informatico-giuridico la distinzione tra sistemi basati su regole e sistemi basati su un modello concettuale del diritto. Usando un linguaggio non privo di connotazioni valutative questo autore ha affermato che primi userebbero una rappresentazione superficiale, i secondi una rappresentazione profonda del diritto<sup>75</sup>. Secondo McCarty,

---

<sup>74</sup>Prendiamo in considerazione soprattutto i sistemi di analisi giuridica.

<sup>75</sup>Cfr. McCARTY T., *Intelligent Legal Information Systems: Problems and Prospects*, cit., p. 266. Per chiarire questa contrapposizione, McCarty ricorre ad un esempio tratto dalla medicina. Il sistema MYCIN è "un sistema superficiale basato su regole, poiché contiene centinaia di regole che collegano i sintomi a infezioni batteriche plausibili, ma non possiede una rappresentazione interna del meccanismo della malattia responsabile di questi sintomi". Invece il sistema CASNET/Glaucoma (cfr. WEISS S., KULIKOWSKI C., AMAREL S., SAFIR A., *A Model-Based Method for Computer-Aided Medical Decision-Making*, in "Artificial Intelligence",



solo costruendo un modello concettuale profondo di determinati settori del diritto sarebbe stato possibile raggiungere i fondamentali obiettivi, pratici e teorici dell'informatica giuridica, cioè:

(a) la realizzazione di un sistema integrato di analisi/pianificazione/reperimento che si avvicini il più possibile al modo nel quale il giurista affronta realmente un problema giuridico e

(b) lo sviluppo di modelli dell'argomentazione giuridica in casi difficili (*hard cases*)<sup>76</sup>.

Recentemente, McCarty ha precisato come segue il concetto di modello concettuale profondo: "vi sono molte categorie del senso comune che stanno alla base della rappresentazione di un settore giuridico: spazio, tempo, massa, azione, permesso, obbligo, causa, scopo, intenzione, conoscenza, credenza e così via. L'idea è selezionare un piccolo insieme di queste categorie del senso comune, quelle che sono più appropriate per una particolare applicazione giuridica e quindi sviluppare un linguaggio per la rappresentazione della conoscenza che rispecchi fedelmente la struttura di questo insieme"<sup>77</sup>. McCarty continua osservando che un linguaggio per la rappresentazione della conoscenza giuridica deve possedere la capacità espressiva necessaria ad esprimere i concetti del senso comune, ma a ciascuna struttura sintattica devono essere associate una semantica e regole inferenza precisamente definite.

Pertanto, il criterio fondamentale per distinguere tra rappresentazioni superficiali e profonde del discorso giuridico, piuttosto che la distinzione tra regole e modelli concettuali<sup>78</sup>, sembra essere quello della profondità della formalizzazione e del trattamento logico. Ricordiamo infatti che un sistema automatico è in grado di "capire" un contesto, nella misura in cui questo sia riformulato in

---

11, pp. 145-172), è "un sistema profondo, o un sistema basato su un modello, poiché le sue diagnosi sono derivate da una rappresentazione dei diversi rapporti gerarchici, causali e temporali che esistono tra gli stati fisiologici associati al glaucoma".

<sup>76</sup>MCCARTY T., *A Language for Legal Discourse. I Basic Features*, relazione presentata al congresso internazionale Artificial Intelligence and Legal Reasoning (Vancouver, giugno 1989), p. 1.

<sup>77</sup>MCCARTY T., *A Language for Legal Discourse*, cit. p. 1.

<sup>78</sup>Non mi sembra esatta la correlazione tra le due distinzioni superficiale-profondo e regole-modello concettuale. Infatti le regole, e in particolare regole formulate nella logica predicativa (le clausole di Horn), sono una struttura sintattica di impiego estremamente generale, che può essere usata anche per esprimere strutture concettuali complesse.

strutture formali manipolabili dal sistema. Quanto meno un contesto è formalizzato, tanto meno esso è "compreso" dal sistema e tanto più è superficiale il suo trattamento automatico.

Il massimo grado di superficialità è raggiunto dai sistemi che si limitano alla logica proposizionale. Nei sistemi di questo tipo (di solito si tratta di sistemi basati su regole) gli enunciati atomici non sono "capiti" dal sistema, che li considera come sequenze arbitrarie di caratteri alfabetici o numerici. La rappresentazione della conoscenza può essere approfondita nelle seguenti direzioni:

-Logica predicativa. Il sistema è in grado di analizzare gli enunciati in elementari in predicati ed argomenti e di "ragionare" sulla base di tale analisi<sup>79</sup>.

-Logiche deontiche, modali, epistemiche. Gli operatori deontici, modali, epistemicci ecc. sono "capiti" dal sistema automatico, nel senso che non sono considerati come sequenze arbitrarie di caratteri, ma sono oggetto di speciali manipolazioni<sup>80</sup>.

-Rappresentazione della struttura concettuale del settore del diritto interessato. In questo caso il sistema è in grado di dedurre nuove informazioni sulla base di rapporti tra i concetti usati (di capire che l'affermazione che usa un concetto più generale copre anche il più specifico ecc.). Un ulteriore approfondimento si avrebbe quando il sistema, comprendesse anche un modello della realtà oggetto della normativa giuridica (rapporti causali, regole di esperienza, correlazioni tra i fenomeni disciplinati ecc.).

La maggior parte dei sistemi informatico-giuridici intelligenti sin qui realizzati sono basati su una rappresentazione superficiale della conoscenza: molti di essi si limitano alla logica proposizionale, alcuni giungono al livello della logica predicativa<sup>81</sup>, nessuno include un trattamento soddisfacente di logiche modali e deontiche, mentre sono rari i sistemi che includano la rappresentazione di rapporti concettuali<sup>82</sup>.

Un interessante tentativo di realizzare una rappresentazione profonda dei concetti giuridici è

---

<sup>79</sup>Cfr. cap. 7 par. 1.2.

<sup>80</sup>Cfr. cap. 7 par. 1.3.

<sup>81</sup>Più spesso si tratta di un sottoinsieme della logica predicativa, in particolare, le clausole di Horn.

<sup>82</sup>Tra i tentativi più interessanti, ricordiamo il progetto LEX e il progetto TAXMAN. Quest'ultimo progetto non era però ispirato ad un approccio di tipo logico.

rappresentato dal *Linguaggio per il discorso giuridico* di McCarty, che però non ha ancora trovato un'implementazione informatica<sup>83</sup>

Sotto un diverso profilo possiamo distinguere tra sistemi che si limitano al ragionamento deduttivo, e sistemi che includono, oltre al ragionamento deduttivo anche la capacità di trattare il ragionamento induttivo e analogico, la capacità di apprendere dall'esperienza ecc.

E' estremamente difficile la valutazione dei sistemi informatico-giuridici sotto questo secondo profilo. Non è stato sviluppato alcun sistema che applichi in modo soddisfacente metodi non deduttivi, e non vi è alcun sistema che unisca il rigore deduttivo all'uso di metodi non deduttivi. Sembra che i due aspetti - il trattamento logico del ragionamento deduttivo, e dello sviluppo di altri tipi di ragionamento del diritto - siano tanto complessi che nessun progetto è riuscito ad approfondire entrambi. Gli studiosi di ispirazione logica si sono soffermati sul primo, altri hanno preferito passare al secondo senza attendere la soluzione del primo. Speriamo che in futuro sarà possibile realizzare, unendo diverse esperienze, sistemi che uniscano al rigore del ragionamento logico, l'impiego usare inferenze non deduttive per collaborare con l'ingegnere della conoscenza nel completamento della base di conoscenza in relazione alle particolarità dei singoli casi<sup>84</sup>.

## **8. Il contributo della filosofia e della teoria del diritto**

La ricostruzione delle strutture fondamentali della conoscenza giuridica, necessaria ai fini dello sviluppo di un sistema intelligente, si avvicina a compiti tradizionalmente attribuiti alla filosofia e alla teoria del diritto. La filosofia e la teoria del diritto sono chiamate a dare un contributo fondamentale nella realizzazione di sistemi basati sulla conoscenza giuridica.

Come abbiamo accennato, ad un sistema esperto debbono essere trasmesse non solo le conoscenze che vengono normalmente comunicate a chi apprende una certa scienza o tecnica, ma anche conoscenze ulteriori, che spesso non è necessario formulare o insegnare espressamente. Si tratta infatti di conoscenze implicite nel senso comune, o nell'esperienza che si connette allo svolgimento di particolari funzioni. E' frequente l'osservazione della

---

<sup>83</sup>Cfr. cap. 8 par. 2.3.

<sup>84</sup>cfr. cap. 4, par. 4.1.

scarsa rilevanza degli studi teorici per la pratica del diritto: l'operatore (ma anche lo scienziato del diritto), si muovono con sicurezza, anche in mancanza di una base teorica. Ciò deriva dal fatto che il ragionamento giuridico è profondamente immerso nel ragionamento comune e nel mondo dell'esperienza. Di conseguenza, l'analisi dei metodi del ragionamento giuridico (e, in particolare, una rappresentazione formale)<sup>85</sup> si rivela, al tempo stesso, scarsamente utile per la pratica e infinitamente complessa.

Nella realizzazione di sistemi informatico-giuridici intelligenti, sistemi in grado di compiere veloci manipolazioni simboliche in base a regole predefinite, ma privi di "buon senso", gli studi metodologici, e non solo quelli che si avvalgono di strumenti formali, rivestono un notevole interesse.

Gli studi di logica delle norme hanno fondamentale importanza, sia per la realizzazione di una base di conoscenza giuridica, sia per la realizzazione del relativo motore inferenziale.

Gli studi in materia di ragionamento giuridico non deduttivo (il problema dell'analogia, dei principi) e di interpretazione non sono meno rilevanti, sia per lo sviluppo di tentativi di andare al di là del ragionamento deduttivo, sia per la determinazione dei limiti delle applicazioni giuridiche dell'intelligenza artificiale.

Si è recentemente affermato che, ai fini dello sviluppo di un sistema esperto giuridico, occorrerebbe cercare una sorta di minimo comune denominatore tra le diverse teorie del diritto: questo minimo comune denominatore esisterebbe (qualora si escludano certe posizioni "patologiche"), comprenderebbe tutto ciò che è necessario e sufficiente allo

---

<sup>85</sup>Il rapporto tra diritto e senso comune non è privo di rilievo anche al fine della scelta del settore giuridico nel quale sviluppare applicazioni intelligenti. L.T. McCarty, osserva che "è allettante iniziare con il problema giuridico più semplice possibile, come le materie dei corsi del primo anno di law school. Saremmo tentati di investigare casi di aggressioni e percosse, o casi in materia di offerta e accettazione del corso sui contratti del primo anno. Ma questi casi sono 'semplici' per gli studenti di diritto soprattutto perché essi si basano sull'esperienza umana ordinaria .... Paradossalmente i casi che sono più trattabili per un sistema di intelligenza artificiale sono quelli che .. un giurista trova più complessi" (MCCARTY T., *Some Requirements for a Computer-Based Legal Consultant*, Report LRP-TR-8, Laboratory for Computer Science Research, Rutgers University, 1980). Infatti la complessità che un sistema automatico può facilmente affrontare è quella relativa alla specificità, al dettaglio della disciplina, non la difficoltà che dipende dalla vaghezza dei concetti o dal conflitto dei principi, dal rapporto col mondo dell'esperienza.

sviluppo di un sistema esperto giuridico<sup>86</sup>. A mio parere questa prospettiva è inaccettabile, in quanto (a) la discriminazione tra fisiologia e patologia non sembra un criterio accettabile per classificare diverse posizioni filosofiche o scientifiche (b) un minimo comune denominatore tra le posizioni "fisiologiche" non esiste<sup>87</sup>; (c) anche se, con qualche artificio concettuale, si riuscisse ad arrivare a qualche posizione comune, ciò basterebbe a giustificare solo le soluzioni più banali.

Un uso consapevole della teoria del diritto nello sviluppo di un sistema esperto giuridico dovrebbe invece porsi il compito di chiarire le scelte metodologiche (ma anche politico-giuridiche) fondamentali che stanno alla base dello sviluppo di un sistema basato sulla conoscenza giuridica, e di fornire strumenti che consentano di realizzare coerentemente tali scelte. Per lo sviluppo di sistemi basati sulla conoscenza giuridica, possono fornire indicazioni importanti sia metateorie del diritto di tipo descrittivo, sia metateorie del diritto di tipo prescrittivo, sia approcci realistici sia approcci normativistici, sia approcci formalistici, sia approcci antiformalistici. Quale che sia l'approccio prescelto, bisogna evitare la confusione concettuale che nasce dalla ricerca di un'uniformità impossibile e deresponsabilizzante.

---

<sup>86</sup>Questa è la tesi fondamentale dalla quale muove SUSSKIND R.E., *Expert Systems in Law*, cit.

<sup>87</sup>Susskind raggiunge questo risultato comparando citazioni tratte da diversi autori senza indagare se sotto talune uniformità terminologiche si nasconda un'uniformità dei concetti usati. Invece, anche se Kelsen e Ross attribuiscono alla scienza giuridica il compito della descrizione del diritto, ciò non significa che per "descrizione", per "diritto", per "scienza giuridica" essi intendano la stessa cosa (come è noto, i due principali teorici del diritto del nostro secolo movevano invece l'uno da una posizione normativistica, l'altro da una prospettiva realistica, che li conduceva a diverse ricostruzioni concettuali (cfr. PATTARO E., *Lineamenti per una teoria del diritto*, cit.); anche se Alchourron e Bulygin parlano di "interpretazione" come di deduzione logica delle conseguenze di un sistema assiomatico, ciò non significa che altri autori che pure parlano di interpretazione, stiano usando lo stesso concetto (cfr. par.4.1).

1. L'assiomatizzazione del diritto	128
2. Scienze naturali e scienze giuridiche	132
3. Ragionamento giuridico e metodo assiomatico	136
3.1. Giustificazione interna e giustificazione esterna della decisione giuridica	138
3.2. Modelli dirazionalità e di giustificazione nei sistemi informatico-giuridici	138
3.3. Relatività delle premesse prescrittive e controllo dei sistemi basati sulla conoscenza giuridica	142
3.4. Conclusione. Ragionamento giuridico e razionalità dei sistemi informatico-giuridici intelligenti	144
4. Interpretazione del diritto e rappresentazione della conoscenza	145
4.1. Il concetto di interpretazione	145
4.2. Interpretazione e testi	148
4.3. Interpretazione e competenze dell'ingegnere della conoscenza giuridica	151
4.4. Il problema dell'interpretazione del fatto	152
5. Conoscenze giuridiche pubbliche e private	155
6. Conoscenze e posizioni giuridiche	157
7. Conoscenze profonde e superficiali	157
8. Il contributo della filosofia e della teoria del diritto	160

## CAP. 7

# METODI E LINGUAGGI PER LA RAPPRESENTAZIONE DELLA CONOSCENZA NELL'INTELLIGENZA ARTIFICIALE

Nelle pagine che seguono saranno illustrate alcune delle tecniche più frequentemente usate nell'intelligenza artificiale per la rappresentazione della conoscenza. Ci si limiterà ad alcuni metodi e linguaggi che assumono un valore paradigmatico: la logica, la programmazione logica, le regole di produzione, le reti semantiche e i frames. Queste tecniche non esauriscono la problematica della rappresentazione della conoscenza: sono stati proposti anche metodi diversi, e gli stessi metodi qui considerati sono state applicati in modi diversi, che meriterebbero distinta considerazione. Ai nostri fini è sufficiente una introduzione superficiale. Un'analisi più dettagliata di alcuni approcci sarà riportata nel capitolo seguente, dedicato alla rappresentazione della conoscenza giuridica.

### 1. La logica

La presentazione dei metodi per la rappresentazione formale della conoscenza non può che partire che dalla logica, il linguaggio formale più ricco di tradizione. Nelle pagine che seguono ci limitiamo ad introdurre brevemente la logica enunciativa, la logica predicativa del primo ordine, la logica modale. Per i necessari approfondimenti si rinvia ad opere specifiche<sup>1</sup>.

---

<sup>1</sup>Tra le numerose introduzioni alla logica cui far riferimento, per approfondire le nozioni che saranno introdotte nelle pagine seguenti, richiamo le seguenti, in ordine crescente di difficoltà:

(a) COPI I., *Introduzione alla logica*, Il Mulino, Bologna, 1964 (traduzione di *Introduction to Logic*, Mac Millan, New York, 1961);

(b) BENCIVENGA E., *Introduzione ai metodi della logica contemporanea*, Boringhieri, Torino, 1984;

(c) QUINE V. O., *Methods of Logic*, Routledge and Kegan Paul, London, 1974 (prima ed. 1952), pp. 280

(d) HODGES W., *Logica*, Garzanti, Milano, 1986, (traduzione di *Introduction to Elementary Logic*, Penguin Books, Harmondsworth;

(e) KALISH D., MONTAGUE R., *Logic. Techniques of Formal Reasoning*, Harcourt, Brace & World, New York, 1974;

### 1.1. La logica enunciativa

La logica enunciativa (o proposizionale) considera gli enunciati come le parti elementari del linguaggio. Pertanto, essa comprende (a) lo studio delle argomentazioni (delle inferenze) basato sulla considerazione della loro struttura enunciativa, cioè della combinazione degli enunciati che le compongono e (b) lo studio del modo in cui la conoscenza può essere rappresentata combinando enunciati. Il linguaggio della logica enunciativa consiste di enunciati e di connettivi logici (simboli che specificano i rapporti tra gli enunciati). La sintassi specifica come combinare enunciati e connettivi per costruire enunciati complessi o molecolari.

Gli enunciati non analizzabili in una combinazione di altri enunciati mediante connettivi logici sono chiamati enunciati atomici. Di fronte a questi enunciati la logica enunciativa si arresta, li considera come unità non ulteriormente analizzabili, le cui caratteristiche non sono rilevanti per l'analisi logica. Infatti, la logica enunciativa si ferma all'esterno degli enunciati e, pertanto, consente solo una rappresentazione superficiale della conoscenza<sup>2</sup>.

Nella logica enunciativa possiamo rappresentare interi enunciati mediante un'abbreviazione, cioè con una lettera alfabetica, o una breve sequenza di caratteri alfabetici. Ciò è possibile proprio in quanto la struttura interna, il contenuto del singolo enunciato è irrilevante. Possiamo stabilire con un glossario la corrispondenza tra abbreviazione ed enunciato. Ad esempio:

**p: Massimo è italiano**

**q: Massimo risiede in Italia**

---

(f) MENDELSON E., *Introduzione alla logica matematica*, Boringhieri, Torino, 1972 (traduzione di *Introduction to Mathematical Logic*, D. Van Nostrand, Princeton (NJ), 1964).

(g) VAN DALEN D., *Logic and Structures*, Springer, Berlin, 1980.

(h) BELL J.L., MACHOVER M., *A Course in Mathematical Logic*, North Holland, Amsterdam, 1974.

Gli ultimi tre manuali approfondiscono gli aspetti formali mentre i testi di Quine, Hodges e Kalish-Montague trattano particolarmente il problema del passaggio dal linguaggio naturale ai formalismi logici, un aspetto particolarmente importante per la rappresentazione della conoscenza giuridica. Una introduzione completa alle problematiche della logica moderna può trovarsi nel manuale in 4 volumi (tre dei quali già disponibili) GABBAY D.M., GUENTHNER F. (a cura di), *Handbook of Philosophical Logic*, Reidel, Dordrecht, 1983-1986.

<sup>2</sup>Sul concetto di rappresentazione superficiale e profonda, cfr. cap. 6 par. 7.



I connettivi logici più usati corrispondono alle parole italiane non, e, o, se-allora, e sono i seguenti

-  $\neg$ , rappresenta la negazione (non):  $\neg p$  significa "Massimo non è italiano".

-  $\&$ , che rappresenta la congiunzione (e):  $p \& q$  significa "Massimo è italiano o Massimo risiede in Italia,

-  $\vee$ , che rappresenta la disgiunzione debole, cioè il vel latino:  $p \vee q$  significa Massimo è italiano o Massimo risiede in Italia.

-  $\rightarrow$ , che rappresenta il condizionale (se-allora):  $p \rightarrow q$  significa "se Massimo è italiano allora Massimo risiede in Italia".

-  $\leftarrow$ , che rappresenta il condizionale rovesciato (se):  $p \leftarrow q$  significa "Massimo è italiano se Massimo risiede in Italia".

-  $\leftrightarrow$ , che rappresenta il doppio condizionale (se e solo se):  $p \leftrightarrow q$  significa "Massimo è italiano se e solo se Massimo risiede in Italia".

I connettivi logici<sup>3</sup> corrispondono a determinate espressioni del linguaggio naturale. Però, mentre le espressioni del linguaggio naturale sono ambigue<sup>4</sup>, i connettivi logici hanno un unico, preciso significato<sup>5</sup>.

Sono stati sviluppati numerosi calcoli formali per la dimostrazione logica, sistemi il cui scopo è ridurre il

<sup>3</sup>Chiamati anche operatori booleani, dal nome di A. Boole, uno degli studiosi che più hanno contribuito allo sviluppo della logica moderna.

<sup>4</sup>Un connettivo logico può corrispondere a più espressioni del linguaggio naturale (ad esempio, la congiunzione logica  $\&$  può tradurre espressioni come "e", "ma", però ecc.), mentre un'unica espressione del linguaggio naturale può corrispondere a diversi connettivi logici (ad esempio, l'espressione "se", può esprimere sia all'implicazione che alla doppia implicazione, l'espressione "o" può avere un senso disgiuntivo o un senso inclusivo ecc.). Il rapporto tra connettivi logici e espressioni del linguaggio naturale è stato approfondito nell'informatica giuridica da L. A. Allen (cfr. i numerosi gli scritti di Allen che affrontano questo problema cfr. ALLEN L.E., *Towards a Normalized Language to Clarify the Structure of Legal Discourse*, cit.).

<sup>5</sup>Nella logica classica, il significato dei connettivi viene specificato associando ad ogni connettivo una "funzione di verità". Ogni connettivo logico consente di costruire, a partire da uno o più enunciati, un enunciato composto, e una regola determinata, associata al connettivo, determina la verità o la falsità dell'enunciato composto a seconda della verità o della falsità degli enunciati componenti. La congiunzione  $p \& q$  è vera se e solo se gli enunciati componenti  $a$  e  $b$  sono entrambi veri; la disgiunzione  $p \vee q$  è vera se e solo se almeno uno degli enunciati componenti è vero; la negazione  $\neg p$  è vera se e solo se  $p$  è falso; il condizionale (materiale)  $p \rightarrow q$  è vero se e solo se  $p$  è falso o  $q$  è vero.

ragionamento a precise regole meccaniche. Una volta che la conoscenza sia stata formalizzata nella logica enunciativa, possiamo usare queste regole per dedurre nuovi enunciati.

Tra i diversi calcoli formali, il più intuitivo è probabilmente il c.d. calcolo della deduzione naturale<sup>6</sup>. "Una prova nel calcolo della deduzione naturale è un albero di formule con un'unica formula alla radice. Le formule in cima ai rami sono chiamate gli assunti della prova. Alcune degli assunti possono essere annullati o scaricati mettendole tra parentesi quadre []. Le premesse della prova sono i suoi assunti non cancellati e la conclusione della prova è la formula alla radice"<sup>7</sup>. Per ciascun connettivo introduciamo due regole di deduzione, una regola per l'introduzione e una regola per l'eliminazione del connettivo. Qui ci limitiamo alle regole per i connettivi &, n- e ->, nella logica classica.

*Regola per l'introduzione della congiunzione (&I);*

$$\frac{A \quad B}{A \& B}$$

cioè, da due formule A e B si può dedurre A & B. Ad esempio da p & q ("Massimo risiede in Italia" & "Massimo è italiano"), possiamo dedurre p ("Massimo risiede in Italia").

*Regola per l'eliminazione della congiunzione (&E);*

$$\frac{A \& B}{A} \qquad \frac{A \& B}{B}$$

cioè dalla formula A & B possiamo dedurre sia la formula A, che la formula B.

*Regola per l'introduzione del condizionale (->I);*

<sup>6</sup>Come è noto, questo approccio fu sviluppato da Genzen (cfr. GENZEN G., *Untersuchungen über das logische Schliessen*, "Mathematische Zeitschrift", 39, pp. 176-210, 405-431, 1934). Tra i testi di logica che usano la deduzione naturale per introdurre la logica classica, cfr VAN DALEN D., *Logic and Structures*, cit.

<sup>7</sup>HODGES W., *Elementary Predicate Logic*, in GABBAY D.M., GUENTHNER F. (a cura di), *Handbook of Philosophical Logic. Vol. 1. Elements of Classical Logic*, Reidel, Dordrecht, p. 29.

$$\begin{array}{c}
 [A] \\
 \cdot \\
 \cdot \\
 \cdot \\
 B \\
 \hline
 A \rightarrow B
 \end{array}$$

cioè, se abbiamo una prova di  $A$  partendo da certi assunti, allora possiamo dedurre  $A \rightarrow B$  partendo da quegli assunti e cancellare l'assunto  $A$  (ad esempio, se ho dimostrato a partire dalle premesse  $p$  e  $q$ , la conclusione  $r$  ("Massimo deve pagare l'imposta sul reddito delle persone fisiche), possiamo dedurre dalla sola premessa  $p$  la conclusione  $q \rightarrow r$  (se Massimo risiede in Italia allora Massimo deve pagare l'imposta sul reddito delle persone fisiche).

*Regola per l'eliminazione del condizionale ( $\rightarrow E$ ) o modus ponens;*

$$\begin{array}{cc}
 A & A \rightarrow B \\
 \hline
 B
 \end{array}$$

cioè dalle formula  $A$  e  $A \rightarrow B$  possiamo dedurre la formula  $B$ .

Vi è anche una regola per l'eliminazione della negazione. Premettiamo che nel nostro calcolo il simbolo  $F$  designa la contraddizione, il falso, l'assurdo. La regola della *reductio ad absurdum* ( $RA$ ) stabilisce se una certa ipotesi  $n-A$  conduce ad una contraddizione, al falso ( $F$ ) allora possiamo dedurre  $A$  e cancellare l'ipotesi  $n-A$ .

$$\begin{array}{c}
 [n-A] \\
 \cdot \\
 \cdot \\
 F \\
 \hline
 A
 \end{array}$$

Vi sono numerosi sistemi di conoscenza che si basano sulla logica enunciativa.

Tra i primi sistemi di questo tipo ricordo il Teorema logico di Newell, Shaw e Simon<sup>8</sup>, che fu usato per dimostrare automaticamente teoremi della logica enunciativa.

<sup>8</sup>Cui si è accennato nel cap. 3 par. 3.1.

I più semplici sistemi basati su regole<sup>9</sup> rappresentano la conoscenza mediante proposizioni ipotetiche al livello della logica enunciativa.

## 1.2. La logica dei predicati del primo ordine

A differenza della logica proposizionale, la logica predicativa si occupa anche della struttura interna degli enunciati atomici (o atomi).

Gli enunciati atomici esprimono proprietà di individui o relazioni tra individui.

Chiamiamo predicati i sintagmi che esprimono le proprietà e le relazioni (grosso modo, i predicati corrispondono ai nomi comuni e ai verbi). Negli enunciati seguenti le parole che compongono i predicati sono sottolineate.

Paolo ama Francesca

Socrate è un filosofo

Massimo è italiano

FIAT è una società commerciale

Le espressioni non sottolineate designano invece gli individui. Queste espressioni vengono chiamate anche termini (e corrispondono, grosso modo, ai nomi propri e ai pronomi). Vi sono tre categorie di termini: costanti, variabili e funzioni.

Chiamiamo costanti i nomi di individui determinati. Ad esempio:

Giovanni, Socrate, informatica, diritto, Bologna, 2.

Chiamiamo invece variabili le espressioni usate per far riferimento ad un individuo qualsiasi.

Scriviamo le variabili con le lettere alfabetiche (minuscole o maiuscole) oppure con una qualsiasi sequenza di caratteri preceduta da \_. Ad esempio:

x, b, y, \_oggetto, \_chiunque.

---

<sup>9</sup>I sistemi basati su regole saranno considerati nel par. 3.

Le *funzioni* consentono di costruire nomi complessi. Ad esempio, il simbolo di funzione **mamma** consente di costruire il nome complesso (la funzione)

**mamma**(Massimo)

(la mamma di Massimo),

Mentre il simbolo di funzione **frazione** consente di costruire la funzione:

**frazione**(3/4)

(la frazione 3/4).

Di solito nella logica gli enunciati atomici vengono espressi in una forma più lontana dal linguaggio naturale: prima viene il predicato (che può essere composto di una sola parola) poi i termini, racchiusi tra parentesi. Ad esempio anziché

**Giovanni risiede a Bologna**

scriveremo

**risiede**(Giovanni, Bologna)

A partire dagli enunciati atomici possiamo costruire enunciati più complessi, grazie ai connettivi della logica enunciativa, e ai quantificatori (il quantificatore universale "per ogni" (**V**) e il quantificatore esistenziale "per qualche" o "esiste un" (**E**)).

Otteniamo le regole di inferenza della logica predicativa aggiungendo le seguenti regole di inferenza a quelle della logica predicativa.

Regola dell'introduzione del quantificatore universale (**VI**) che consente di dedurre un enunciato universale dall'enunciato relativo ad un soggetto arbitrario (designato dalla variabile **x**).

$$\text{VI} \quad \frac{A(x)}{Vx A(x)}$$

Regola dell'eliminazione del quantificatore universale, consente di possiamo dedurre, da un'enunciato universale, enunciati più specifici (che otteniamo eliminando il

quantificatore e sostituendo la variabile  $x$  con il termine  $t$ )<sup>10</sup>.

$$\text{VE} \frac{Vx A(x)}{A(t)}$$

Vediamo un'applicazione della seconda regola.  
L'enunciato :

"se una persona ha il padre italiano, allora essa è italiana.

può essere rappresentato come segue:

$$1. V(x,y) ((\text{padre}(y,x) \ \& \ \text{\_italiano}(y)) \rightarrow \text{\_italiano}(x) .$$

Da questo enunciato e dai seguenti:

$$2. \text{\_italiano}(\text{Luigi})$$

$$3. \text{padre}(\text{luigi}, \text{Mario}) ,$$

possiamo dedurre nella logica predicativa l'enunciato

$$4. \text{\_italiano}(\text{Mario}) .$$

La deduzione potrebbe essere la seguente (abbrevio padre con  $p$ ,  $\text{\_italiano}$  con  $i$ , Luigi con  $L$  e Mario con  $M$ )

$$\begin{array}{c} V(x,y) \ (i(y) \& p(y,x) ) \rightarrow i(x) \qquad i(L) \qquad i(L,M) \\ \text{VE} \frac{\qquad \qquad \qquad}{(i(L) \& p(L,M)) \rightarrow i(M)} \qquad \& I \frac{i(L) \qquad i(L,M)}{i(L) \ \& \ i(L,M)} \\ \qquad \rightarrow E \frac{\qquad \qquad \qquad}{i(M)} \end{array}$$

Se avessimo dovuto rappresentare i medesimi enunciati nella logica enunciativa, ci saremmo dovuti accontentare di una rappresentazione del tipo

<sup>10</sup>La definizione delle regole di inferenza della logica predicativa andrebbe completata con talune specificazioni. In particolare nella regola VI la variabile  $x$  non può comparire libera in alcuna ipotesi dalla quale dipenda  $A(x)$ , cioè in un assunto non cancellato della prova di  $A(x)$ ; nella regola. In VE il termine  $t$ , che sostituiamo ad  $x$  nella formula  $A$  deve essere libero per  $x$  in  $A$ . Una variabile è libera se non è oggetto di quantificazione, se non rientra nel campo d'azione di un quantificatore. Per un approfondimento di queste nozioni, che vanno al di là dei limiti del presente volume, si rinvia ai testi specialistici menzionati nella nota n. 1.

1. "una persona ha il padre italiano"  $\rightarrow$  "essa è italiana".
2. "Luigi è il padre di Mario"
3. "Luigi è italiano"

dalla quale non avremmo potuto inferire formalmente che Mario è italiano.

In molti testi di logica si usa un approccio diverso dalla deduzione naturale. Il calcolo logico, anziché con un insieme di regole di inferenza, viene introdotto mediante un insieme di assiomi e con una o due regole di inferenza (di regola si tratta del *modus ponens*). Gli assiomi sono schemi di formule tautologiche<sup>11</sup>. Le formule tautologiche possono essere usate in qualsiasi deduzione, in quanto deducibili senza premesse. Questo approccio alla logica è meno intuitivo, ma presenta alcuni vantaggi. In particolare, consente di definire facilmente logiche che rappresentino alternative o estensioni della logica classica (basterà modificare gli assiomi). La logica proposizionale classica può essere introdotta con i seguenti quattro assiomi (e la regola del *modus ponens*):

- A1  $(A \vee A) \rightarrow A$
- A2  $A \rightarrow (A \vee B)$
- A3  $(A \vee B) \rightarrow (B \vee A)$
- A4  $(B \rightarrow C) \rightarrow ((A \vee B) \rightarrow (A \vee C))$

Alle regole di inferenza della logica predicativa, si possono far corrispondere degli assiomi:

- A5.  $\forall x A(x) \rightarrow A(t)$
- A6.  $A(x) \rightarrow \forall x A(x)$

La realizzazione di sistemi che usano l'intera logica dei predicati pone notevoli problemi computazionali<sup>12</sup>. I tentativi di sviluppare sistemi che siano in grado di combinare l'espressività della logica dei predicati con la possibilità di un uso efficiente di metodi deduttivi

---

<sup>11</sup>Le verità logiche o tautologie (ad esempio,  $A \vee \neg A$ , nella logica classica) sono enunciati che sono sempre veri, quale che sia il valore di verità degli enunciati atomici che le compongono. Si suole distinguere tra tautologie ("piove e non piove") e verità fattuali o empiriche ("piove"): le prime sono sempre vere, le seconde sono vere o false a seconda di come stanno le cose.

<sup>12</sup>A mia conoscenza, il sistema LEX è l'unico sistema informatico-giuridico nel quale si è cercato di implementare l'intera logica dei predicati (cfr. cap. 4 par. 3.2).

automatici, ha condotto allo sviluppo dei linguaggi di programmazione logica, che illustreremo nel par. 2.

### 1.3. Altre logiche

Nel paragrafo precedente ci si è limitati a presentare la logica classica, l'edificio teorico che fu sviluppato tra la fine dell'ottocento e l'inizio del 900 grazie all'opera di studiosi come Frege, Peano, Russel, Tarski ecc, e che per molto tempo fu ritenuto l'unica logica possibile. Nel nostro secolo si è assistito, tuttavia, alla proliferazione di numerose logiche, alcune delle quali rappresentano estensioni, altre alternative della logica classica. Ad alcune estensioni della logica classica si accennerà nel paragrafo seguente. Tra gli approcci alternativi ricordo la logica intuizionista e le logiche rilevanti, la cui applicazione al ragionamento giuridico, anche al fine della realizzazione di sistemi informatico-giuridici intelligenti è stata suggerita rispettivamente da L.T. McCarty<sup>13</sup> e L.E. Allen<sup>14</sup>.

Qui ci si limiterà ad accennare a logiche modali e deontiche che costituiscono estensioni della logica classica<sup>15</sup>. Si tratta cioè di logiche che si ottengono aggiungendo alla sintassi della logica enunciativa o della logica predicativa (a) i vocaboli che esprimono formalmente i concetti modali o deontici e le regole per usare tali vocaboli nelle formule logiche (b) assiomi o regole di inferenza per l'uso degli operatori modali o deontici.

#### 1.3.1. Le logiche modali

La logica modale formalizza i concetti di necessità (con l'operatore *L*, che significa "è necessario che") e di possibilità (con l'operatore *M*, che significa "è possibile che", e introduce speciali assiomi (o regole di inferenza) che consentono di operare inferenze basate sugli operatori modali, inferenze sarebbero impossibili nella logica

---

<sup>13</sup>Cfr., cap. 8 par. 2.3.

<sup>14</sup>Cfr., ALLEN L.E., SAXON C.S., *Analysis of the Logical Structure of Legal Rules by a Modernized and Formalized Version of Hohfeld's Fundamental Legal Conceptions*, in MARTINO A.A., SOCCI NATALI F. (a cura di) *Automated Analysis of Legal Texts*, cit., pp. 385-450.

<sup>15</sup>Per un'introduzione alle estensioni della logica classica, cfr. GABBAY D.M., GUENTHNER F. (a cura di), *Handbook of Philosophical Logic. Vol. 2. Extensions of Classical Logic*, Reidel, Dordrecht, 1984.



enunciativa e predicativa<sup>16</sup>. Sono stati sviluppati numerosi sistemi di logica modale.

Seguendo von Wright<sup>17</sup>, per logiche modali normali possiamo intendere i sistemi logici che si ottengono aggiungendo alle regole di inferenza della logica enunciativa classica almeno gli assiomi e le regole seguenti:

$$A1. \quad L(A \ \& \ B) \ \leftrightarrow \ L \ A \ \& \ L \ B$$

(E' necessario che A e B se e solo se, è necessario che A ed è necessario che B).

$$A2. \quad LA \rightarrow A$$

(Se A è necessario allora A è vero)

$$A3. \quad Lt$$

Sono necessarie tutte le tautologie (della logica proposizionale).

In aggiunta alle regole di inferenza del calcolo proposizionale serve la c.d. regola di estensionalità, che consente lo scambio di proposizioni logicamente equivalenti nelle formule della logica modale<sup>18</sup>.

Il concetto di possibilità può introdotto con una definizione:

$$MA \text{ =df } n\text{-}L \ n\text{-}A$$

<sup>16</sup>Per un'introduzione alla logica modale, cfr. HUGHES G.E., CRESSWELL M.J., *An Introduction to Modal Logic*, Methuen, London, 1968, pp. 388 (ristampa con correzioni 1972, ristampa 1982); CHELLAS B.F., *Modal Logic. An Introduction*, Cambridge University Press, London, pp. 295, 1980; BULL R.A., SEGERBERG K., *Basic Modal Logics*, in GABBAY D.M., GUENTHNER F. (a cura di), *Handbook of Philosophical Logic. Vol. 2. Extensions of Classical Logic*, Reidel, Dordrecht, 1984, pp. 1-88.

<sup>17</sup>Von WRIGHT H., *Problemi e prospettive della logica deontica. Una panoramica*, 1969, p. 385 s.

<sup>18</sup>La regola di estensionalità e l'assioma A3, consentono di dimostrare la c.d. regola di necessitazione che stabilisce che, se un enunciato A è una tesi (è deducibile senza premessa) allora anche LA è una tesi (cioè che un enunciato logicamente vero è anche necessario). In alcune introduzioni alla logica modale si introduce direttamente questa regola di inferenza.

("è possibile che A" significa per definizione "non è necessario che non A").

Il sistema che comprende solo gli assiomi e le regole di inferenza appena indicati è il c.d. sistema T, sviluppato da Gödel e Feys negli anni 30<sup>19</sup>. Aggiungendo ulteriori assiomi al sistema T, si ottengono altri sistemi modali<sup>20</sup>.

La logica modale consente di compiere inferenze che non sarebbero possibili nella logica enunciativa o predicativa. Ad esempio, l'enunciato seguente:

**"non è possibile che Massimo sia colpevole"**

può essere espresso come segue:

**n-M(colpevole(Massimo)),**

dal quale possiamo inferire, ad esempio

**L n-(colpevole(Massimo))**

("necessario che Massimo non sia colpevole") e quindi

**n-(colpevole(Massimo))**

("Massimo non è colpevole").

Sono stati sviluppati alcuni sistemi informatici in grado di trattare aspetti della logica modale, ma per ora senza risultati operativi.

---

<sup>19</sup>GÖDEL K., *Collected Works*, Oxford University Press, New York, 1986, pp. 301-302 (già pubblicato come *Eine Interpretation des Intuitionistischen Aussagenkalkulus*, in "Ergebnisse eines Mathematischen Kolloquiums", 4, 1933, pp. 34-38); FEYS R., *Les logiques nouvelles des modalités*, in "Revue Néoscholastique de Philosophie", vol. 40, 1937, pp. 517-553.

<sup>20</sup>Tra i più noti, sistemi S4 e S5, che si ottengono aggiungendo a T rispettivamente gli assiomi

**L A → LL A**

(se è necessario che A allora è necessario che sia necessario che A)

e

**M A → LM A**

(se è possibile che A allora è necessario che sia possibile che A).

### 1.3.2. Le logiche deontiche

La logica deontica<sup>21</sup> presenta particolare interesse per gli studi informatico-giuridici, poiché tratta i concetti normativi: il dovere, il permesso<sup>22</sup> e talvolta anche il potere (o l'autorizzazione)<sup>23</sup>. Come è noto, la logica deontica è un settore controverso e ancor oggi, a quarant'anni dalla nascita di questa disciplina<sup>24</sup>, mancano risultati consolidati. Molte logiche deontiche conducono a paradossi, cioè a conclusioni controintuitive e nessuna delle logiche deontiche sinora sviluppate sembra completamente soddisfacente. Le ricerche informatico giuridiche hanno dato nuovo impulso agli studi di logica deontica, e, in particolare, hanno condotto allo sviluppo di logiche deontiche ispirate a considerazioni di efficienza computazionale<sup>25</sup>. Qui non sarà approfondito il tema della logica deontica nell'intelligenza artificiale ma ci si limiterà a presentare il sistema che, secondo von Wright<sup>26</sup>, rappresenta la logica deontica standard.

---

<sup>21</sup>Per una rassegna aggiornata degli studi di logica deontica, cfr. AQUIST L., *Deontic Logic*, in GABBAY D.M., GUENTHNER F. (a cura di), *Handbook of Philosophical Logic. Vol. 2. Extensions of Classical Logic*, Reidel, Dordrecht, 1984, pp. 605-714.

<sup>22</sup>Ai quali si limitano le logiche deontiche standard.

<sup>23</sup>Come nelle logiche che si ispirano ai concetti hohfeldiani.

<sup>24</sup>Benché importanti predecessori della moderna logica deontica si possano ravvisare in filosofi o teorici del diritto come Leibniz, Bentham, Hohfeld, Mally, la nascita di questa disciplina può essere fatta risalire all'inizio degli anni 50, quando furono pubblicati tre fondamentali lavori, elaborati indipendentemente: von WRIGHT, *Deontic Logic*, in "Mind", 60, 1951, pp. 1-15 (traduzione italiana *Logica deontica*, appendice di DI BERNARDO, *Logica, norme, azione*, 1969, pp. 119-148)); BECKER O., *Untersuchungen über den Modalkalkül*, A. Hain, Maisenheim am Glan, 1952; e KALINOWSKI G., *Théorie des propositions normatives in Logique déontique. I* (1953-1969), pp. 147-182 (già in "Studia logica", 1953).

<sup>25</sup>Tra gli studiosi che si sono occupati del problema vanno menzionati soprattutto ALLEN L.E., SAXON C.S., *Analysis of the Logical Structure of Legal Rules by a Modernized and Formalized Version of Hohfeld's Fundamental Legal Conceptions*, cit.; MCCARTY T., *Permission and Obligation: an Informal Introduction*, cit.

<sup>26</sup>Von WRIGHT H., *Problemi e prospettive della logica deontica. Una panoramica*, 1969, p. 385 s.

La logica deontica proposizionale standard si ottiene aggiungendo alla logica proposizionale i seguenti assiomi:

A1.  $O(A \ \& \ B) \leftrightarrow OA \ \& \ OB$

A2.  $n-O \ n-t$

e inoltre la regola di estensionalità (che consente lo scambio di proposizioni logicamente equivalenti nelle formule deontiche) e la seguente definizione:

$PA =df \ n-O \ n-A$

La logica deontica consente operare inferenze che non sarebbero possibili nella sola logica proposizionale o predicativa<sup>27</sup>. Ad esempio si può esprimere formalmente l'enunciato "Giuseppe deve pagare 500.000 a Mario", con l'enunciato seguente:

$O(paga(Giuseppe, Mario, 500.000)),$

dal quale possiamo inferire, in molte logiche deontiche<sup>28</sup>:

$P(paga(Giuseppe, Mario, 500.000))$

(è permesso che Giuseppe paghi 500.000 lire a Mario).

Anche altre logiche, che ci si limita ad elencare, presentano un interesse per le applicazioni giuridiche intelligenza artificiale, come le logiche temporali (che trattano il tempo, formalizzando i concetti di futuro, passato, prima, dopo ecc.), le logiche dinamiche (che si occupano del problema della trasformazione di stati di cose e di valori di verità), le logiche epistemiche (che formalizzano i significati espressi da espressioni come "credo che" o "so che"), le logiche condizionali (che si approfondiscono il problema della rappresentazione delle

---

<sup>27</sup>Resta dubbia, allo stato del dibattito logico filosofico e delle realizzazioni informatiche, la reale utilità di un'implementazione informatica della una logica deontica.

<sup>28</sup>Questa deduzione non è possibile in tutte le logiche deontiche. Ad esempio, non vale nel sistema proposto da MCCARTY T., *Permission and Obligation: an Informal Intruduction*, in MARTINO A.A., SOCCI NATALI F., *Automated Analysis of Legal Texts*, cit., pp. 307-337 (cfr. JONES A.J., *On the Relationship between Permission and Obligation*, in *Proceedings of the First International Conference on Artificial Intelligence and Law* (Boston, maggio 1987), pp. 164-169.

implicazioni), le logiche erotetiche (che hanno per oggetto le domande), le logiche fuzzy (che trattano l'incertezza). Spesso, per il trattamento di questioni dell'intelligenza artificiale, è preferibile o necessario il ricorso a logiche di ordine superiore, cioè a logiche che consentono di trattare anche predicati o proposizioni come individui (e quindi di farli oggetto di quantificazione).

## 2. La programmazione logica

R.A. Kowalski, uno dei fondatori della programmazione logica<sup>29</sup> introduce così questa disciplina: "La programmazione logica può essere definita, in generale, come l'uso della logica simbolica per la rappresentazione esplicita di problemi e delle basi di conoscenza associate ad essi, insieme con l'uso di inferenze logiche controllate per la soluzione effettiva di questi problemi. Attualmente, la programmazione logica è generalmente intesa in termini più specifici: il linguaggio per la rappresentazione della conoscenza è un sottoinsieme particolare (le clausole di Horn) della logica predicativa classica del primo ordine e il meccanismo per la soluzione dei problemi è una forma particolare (la risoluzione) di inferenza classica del primo ordine"<sup>30</sup>.

La programmazione logica ebbe origine da due linee di ricerca convergenti:

- rendere la logica più pratica, farne lo strumento per risolvere concreti problemi computazionali;

- dotare l'informatica (la programmazione) della capacità espressiva e del rigore della logica.

Le basi teoriche della programmazione logica si trovano in alcuni fondamentali risultati nel campo della dimostrazione automatica di teoremi ottenuti già negli anni

---

<sup>29</sup>Per un'introduzione generale alla programmazione logica, cfr. KOWALSKI R.A., *Logic for Problem Solving*, North Holland, New York, Oxford, 1979, pp. 287; HOGGER J.C., *Introduction to Logic Programming*, Academic Press, London, 1984; LLOYD J.W., *Foundations of Logic Programming*, Springer, Berlin, 1984 (trad. it. *Fondamenti della programmazione logica*, Muzzio, Padova, 1986). Per un'introduzione al PROLOG, cfr., tra i numerosi sull'argomento, CLOCKSIN W.F., MELLISH C.S., *Programming in Prolog*, Springer Verlag, New York, 1981; CASADEI G., TEOLIS A., *Prolog. Dalla programmazione all'intelligenza artificiale*, Bologna, Zanichelli, 1986; WALKER A., MCCORD M., SOWA J.F., WILSON W.G., *Expert Systems and Prolog*, cit.; STERLING L., SHAPIRO E., *The Art of Prolog*, cit., 1987.

<sup>30</sup>KOWALSKI R.A., *Logic Programming*, in SHAPIRO S.C., ECKROTH D. (a cura di), *Encyclopedia of Artificial Intelligence*, 1987, p. 544.

sessanta, e in particolare, nella risoluzione di J.A. Robinson<sup>31</sup> e nel metodo dell'eliminazione del modello di D.W. Loveland<sup>32</sup>.

Tuttavia la nascita di questo nuovo approccio alla programmazione risale solo all'inizio degli anni 70, ed è dovuta alla collaborazione di tre studiosi, R. Kowalski dell'università di Edimburgo e C. Colmerauer e P. Roussel dell'università di Marsiglia. Questi studiosi svilupparono i concetti fondamentali della programmazione mediante clausole di Horn<sup>33</sup>, e discussero l'applicazione della risoluzione al problema dell'analisi del linguaggio naturale. Colmerauer e Roussel completarono nel 1972 l'implementazione del primo interprete PROLOG (*PROgrammation en LOGique*), dopo aver realizzato un sistema per l'interazione con sistemi informatici in linguaggio semi-naturale (in francese)<sup>34</sup>.

La programmazione logica muove dall'assunto che la conoscenza possa essere espressa in linguaggi logici e che la risposta a problemi o quesiti determinati possa essere ottenuta mediante algoritmi (motori inferenziali) riconducibili al modello della dimostrazione di teoremi. Pertanto nella programmazione logica appare con la massima chiarezza il modello del sistema assiomatico che, a nostro parere, caratterizza le applicazioni di intelligenza artificiale<sup>35</sup>.

Nella programmazione logica si ricorre di regola, anziché alla logica predicativa standard<sup>36</sup>, a formalismi che si richiamano ad un sottoinsieme della logica predicativa, la forma clausale<sup>37</sup>.

---

<sup>31</sup>ROBINSON J.A., *A Machine-Oriented Logic Based on the Resolution Principle*, in JACM, 12, 1965, pp. 23-41.

<sup>32</sup>LOVELAND D.W., *Mechanical Theorem Proving by Model Elimination*, in JACM, 15, 1968, pp. 223-251.

<sup>33</sup>KOWALSKI R.A., *The Predicate Calculus as a Programming Language*, in *Proceedings of the First Symposium on the Mathematical Foundations of Computer Science*, Jablonna (Polonia), 1972

<sup>34</sup>COLMERAUER A., KANOUI H., PASERO R., ROUSSEL P., *Un système de communication homme-machine en Français*, relazione interna, Groupe d'intelligence artificielle, Université d'Aix-Marseille, Luminy, France, 1972.

<sup>35</sup>Cfr. cap. 1, par. 3.

<sup>36</sup>La sintassi che si è introdotta informalmente nel par. 1.2.

<sup>37</sup>Una clausola nella forma disgiuntiva (comunemente usata dai logici) è una disgiunzione di enunciati atomici della logica predicativa o di loro

Una clausola, in generale, può avere la struttura seguente:

**testa :- corpo**

La testa è un enunciato atomico, o una disgiunzione di enunciati atomici, il corpo può essere una singolo enunciato atomico o una congiunzione di enunciati atomici. Il corpo o la testa possono mancare.

L'uso della logica predicativa del primo ordine, anche nella forma clausale, pone notevoli problemi computazionali. Pertanto, i linguaggio logico più usati per sviluppare applicazioni informatiche si richiamano tutti alla programmazione logica in senso stretto, che si basa su un sottoinsieme della logica clausale, le clausole di Horn (le clausole che hanno al massimo un conseguente). Nel seguito ci si limiterà a considerare clausole di Horn.

Possiamo avere:

1. Clausole senza testa, che rappresentano quesiti o mete (goals). Ad esempio <-FIAT è una persona giuridica rappresenta il quesito "la FIAT è una persona giuridica?".

2. Clausole senza corpo, dette clausole unitarie (unit clauses), che esprimono affermazioni incondizionate (ad esempio, IBM è un società per azioni;

3. clausole con corpo e testa, dette clausole non unitarie, che esprimono affermazioni condizionali. Ad esempio, x è una persona giuridica <- x è un società per azioni.

Le clausole con un conseguente (quelle sub 2 e 3) si dicono anche clausole definite.

Gli enunciati atomici si esprimono come nella logica predicativa.

Il nucleo logico del PROLOG si ottiene aggiungendo alle clausole di Horn la c.d. negazione come fallimento (negation as failure). Adottare la negazione come fallimento significa assumere che l'insuccesso (il fallimento) nella prova di un

---

negazioni. Le variabili che compaiono nella clausola si intendono quantificate universalmente. E' possibile dimostrare che qualsiasi conoscenza esprimibile nella logica dei predicati del primo ordine può essere riformulata nella forma clausale.

Di regola, i sistemi di programmazione logica consentono di usare un sintassi più leggibile della forma disgiuntiva. In particolare, vengono omessi i quantificatori universali, e le negazioni di enunciati atomici vengono riscritti come conclusioni affermative di implicazioni ( $n-A \vee B$  equivale a  $A \leftarrow B$ ). In questo modo si ottiene la sintassi cui faremo riferimento nel seguito. Ad esempio, le clausole:

**$\forall x \text{ non}(\text{mortale}(x) \vee \text{uomo}(x))$**

("per ogni x, x è un uomo oppure x è mortale) diventa la clausola

**$\text{mortale}(x) \leftarrow \text{uomo}(x)$ ,**

("x è mortale, se x è un uomo").

enunciato equivalga alla negazione dello stesso enunciato. Si assume cioè che valga  $n\text{-}P$  se non è possibile provare  $P$  in un tempo finito. Si è dimostrato che la negazione come fallimento è un'adeguata approssimazione al significato della negazione logica solo quando si muova dalla c.d. *assunzione del mondo chiuso*, cioè dall'ipotesi che il programma (la base di conoscenza) contenga una completa descrizione delle proprietà e delle relazioni designate dai predicati che vi compaiono<sup>38</sup>.

Ad esempio, supponiamo che la base di conoscenza del sistema comprenda solo gli enunciati

Massimo è italiano  
Laura è italiano<sup>39</sup>

Da questa base di conoscenza è deducibile che non (Carlo è italiano), in quanto l'enunciato Carlo è italiano non è dimostrabile. Questa conclusione non è logicamente valida<sup>40</sup> (se non si accolga l'assunto del mondo chiuso) poiché le premesse non escludono che Carlo possa essere italiano.

## 2.1. IL PROLOG: la rappresentazione della conoscenza

Nel PROLOG possiamo usare le clausole definite (cioè le clausole che hanno una conclusione, non i quesiti) per rappresentare la conoscenza. Un programma logico (una base di conoscenza) è un insieme di assunti espressi in forma di clausole:

(a) Le clausole unitarie sono affermazioni incondizionate, o fatti. Un fatto è l'affermazione che un'entità ha una certa proprietà o che sussiste una certa relazione tra due o più entità. Ad esempio

Gino è italiano  
Gino è genitore di Carlo.

Possiamo usare clausole unitarie anche per fare affermazioni generali, per esprimere fatti universali, ad esempio:

---

<sup>38</sup>Cfr. CLARK K.L., *Negation as Failure*, in GALLAIRE H., MINKER J. (a cura di) *Logic and Data Bases*, Plenum Press, New York, 1978, pp. 293-322

<sup>39</sup>Il predicato è italiano resta lo stesso sia per Laura che per Massimo.

<sup>40</sup>Una inferenza è logicamente valida se è impossibile che le premesse siano vere e la conclusione falsa.



x è uguale a x

("ogni cosa è uguale a sé stessa"),

x ama denaro),

("tutti amano il denaro").

(b) Le clausole non unitarie sono affermazioni condizionate o regole. Le clausole non unitarie si scrivono come implicazioni rovesciate, nelle quali cioè il simbolo dell'implicazione punta a sinistra e la conclusione va scritta prima delle condizioni<sup>41</sup>.

Ad esempio, dati i fatti:

Olivetti è una società per azioni

Ibm è una società per azioni

la regola

x è una persona giuridica <- x è una società per azioni.

consente di inferire i fatti Olivetti è una persona giuridica e IBM è una persona giuridica, senza dichiararli espressamente nel programma.

Una conclusione può essere subordinata a più condizioni: in questi casi usiamo regole che abbiano una congiunzione di condizioni:

x è padre di y <- x è genitore di y & x è maschio

(x è padre di y, se x è genitore di y e x è maschio).

## 2.2. I quesiti

La base di conoscenza (le regole e i fatti) per poter essere attivati debbono essere caricati nello spazio di lavoro (*working space*)<sup>42</sup> del sistema PROLOG. Affinché la conoscenza venga usata, affinché il motore inferenziale inizi ad operare, occorre un'ulteriore condizione. Deve

---

<sup>41</sup>Sono state sviluppate numerose implementazioni del PROLOG, e non tutte adottano i simboli usati di solito nella logica (che si useranno anche nelle pagine seguenti). Nella sintassi più diffusa, la c.d. sintassi di Edimburgo, il condizionale rovesciato si rappresenta con il simbolo :- la congiunzione si scrive con la virgola e la disgiunzione con il punto e virgola.

<sup>42</sup>Facciamo riferimento al funzionamento del Prolog interpretato.

essere proposto al sistema un quesito (*query*) o, come anche si dice considerando che la risposta al quesito è l'obiettivo che il sistema deve raggiungere, una meta (*goal*). Dal punto di vista della sintassi i quesiti semplici non differiscono dai fatti. Come un fatto, anche un quesito semplice deve essere formulato come un enunciato atomico<sup>43</sup>. Di regola il sistema interpreta come fatti gli enunciati compresi nel programma e come quesiti quelli digitati alla tastiera dopo che il programma (la base di conoscenza) è stato caricato nella memoria centrale dell'elaboratore<sup>44</sup>.

Possiamo distinguere tre tipi fondamentali di quesiti<sup>45</sup>:

1. *quesiti di verifica*, cioè domande del tipo sì o no, che chiedono se un certo predicato è vero, cioè se una particolare proprietà si addice ad un'entità o se una certa relazione sussiste tra più entità. Nelle domande di questo tipo non compaiono variabili (libere). Ad esempio, la domanda Socrate è mortale chiede se Socrate è un mortale, esige una risposta affermativa o negativa.

2. *Quesiti di ricerca*, corrispondenti alle domande in italiano che usano i pronomi interrogativi "chi" o "che cosa". Un quesito di questo tipo contiene variabili (libere) e viene interpretato come la richiesta di trovare un valore per tali variabili. Ad esempio, la domanda x è mortale richiede al sistema di trovare un ente che sia mortale e di legare a (cioè sostituire con) il suo nome la variabile x. La risposta potrà essere x=Socrate. Analogamente, la domanda x è genitore di Carlo, va letta come "chi è genitore di Carlo?" (in logica, "esiste un x tale che x sia genitore di Carlo") e la risposta potrà essere x=Gino.

3. *Quesiti di azione*, che chiedono al sistema di eseguire operazioni extralogiche di vario tipo. Non si tratta di quesiti che vertono sulla base di conoscenza, ma di richieste di causare effetti collaterali (ad esempio, di modificare la base di conoscenza aggiungendo o togliendo fatti, di effettuare operazioni di input-output ecc.)<sup>46</sup>.

---

<sup>43</sup>Possiamo formulare anche domande complesse, che esprimiamo con come una congiunzione di enunciati atomici, eventualmente negativi.

<sup>44</sup>Spesso, i sistemi PROLOG consentono di indicare se un enunciato compreso nel programma debba considerarsi come un quesito o se all'opposto, un enunciato digitato alla tastiera debba considerarsi come un fatto.

<sup>45</sup>Riprendo questa distinzione, abituale nei testi dedicati al PROLOG, nei termini in cui è formulata da SOWA J.F., *A Prolog to Prolog*, cit., p. 32.

<sup>46</sup>Non ci si sofferma su questo tipo di operazioni, per le quali si fa rinvio ai manuali sul PROLOG.

La trasparenza logica è pregiudicata dall'uso di predicati extralogici<sup>47</sup>. Consideriamo, ad esempio, i predicati extralogici `read` (predicato di input), e `write` (predicato di output) che esprimono (considerando i dispositivi standard di input/output) la richiesta di trasferire un certo input dalla tastiera allo spazio di lavoro e di causare un certo output al video. La clausola logica

1. `x è cittadino italiano se`  
`x è nato in Italia.`

potrebbe diventare la clausola

2. `x è cittadino italiano se`  
`write("scrivi il nome della persona:?" )`  
`read(x)`  
`write(x è nato in Italia?),`  
`read(y),`  
`y="si"`  
`write(confermo che x è cittadino italiano),`

nella quale la struttura logica è divenuta irriconoscibile.

A questo problema si può porre rimedio mediante l'uso di strumenti per l'interrogazione dell'utente (*query the user*)<sup>48</sup>. Se il sistema comprende strumenti di questo tipo, l'ingegnere della conoscenza potrà limitarsi alla redazione della clausola n. 1 e sarà il sistema che, per rispondere alla domanda

**Carlo è cittadino italiano?**

porrà la domanda

**'E' vero che Carlo è nato in Italia ?,**

e accetterà risposte del tipo

**Carlo è nato in Italia.**

In questa prospettiva "la base di conoscenza totale usata per rispondere al quesito comprende non sole le clausole del programma sorgente ma anche le clausole.

---

<sup>47</sup>KOWALSKI R.A., *Logic Programming*, cit., p. 553

<sup>48</sup>Strumenti di questo tipo sono stati implementati in alcuni sistemi PROLOG e, in particolare, nel guscio APES, usato nelle applicazioni giuridiche presso l'Imperial College (cfr. cap. 8 par. 2.1).

fornite dall'utente... La base di conoscenza è vista come condivisa tra l'utente e il sistema, ciascuno dei quali può interrogare l'altro per determinare la base di conoscenza e le sue conseguenze logiche"<sup>49</sup>.

L'uso di strumenti di questo tipo, come si vedrà, è particolarmente importante nel diritto, in quanto consente al giurista di concentrarsi sul problema della rappresentazione della conoscenza, senza perdersi nei dettagli del funzionamento dell'interfaccia con il sistema.

### 2.3. Il motore inferenziale.

La proposizione di un quesito attiva uno speciale programma, il motore inferenziale del PROLOG.

Nell'esaminare come opera il motore inferenziale si passa da una considerazione dichiarativa ad una considerazione procedurale del programma logico. Gli enunciati della base di conoscenza non ci appaiono più (solo) come l'enunciazione di certi assunti, di certe verità, ma piuttosto come procedure da eseguire per ottenere un certo risultato.

La clausola

**A <- B & C & ...**

va letta come la procedura seguente: per risolvere il problema (o raggiungere la meta) A, risolvi i sottoproblemi (raggiungi le mete) B, C, ...

Ad esempio,

**x è italiano <- x è genitore di y & y è italiano.**

va letta come segue: per determinare se x è italiano, determina se y è genitore di x, e determina se y è italiano.

Possiamo avere più regole con la stessa conclusione, cioè più modi per risolvere lo stesso problema.

Ad esempio, per determinare se una persona è italiana potremmo applicare anche la regola:

**x è italiano <- x è coniuge di y & y è italiano.**

I sistemi PROLOG eseguono le procedure di cui si tratta in un ordine semplice e definito. I sottoproblemi (le sottomete) cioè le condizioni cui una clausola subordina una certa conseguenza vengono affrontati nell'ordine da sinistra a destra; le diverse regole (procedure) per risolvere uno

---

<sup>49</sup>KOWALSKI R.A., *Logic Programming*, cit., p. 553.

stesso problema vengono esaminate seguendo l'ordine dall'alto in basso<sup>50</sup>.

Vi sono due tipi fondamentali di ragionamento che possono essere usati per dedurre delle conclusioni da un insieme di premesse espresse in forma di clausole: il ragionamento in avanti e il ragionamento all'indietro.

Il ragionamento in avanti inizia esaminando i fatti compresi nelle premesse. Usa le regole per dedurre nuovi fatti, e si arresta quando trova un fatto che corrisponde al quesito proposto. Questo metodo di ragionamento presenta l'inconveniente che esso non è indirizzato con precisione dal problema da risolvere: prima di giungere alla risposta può essere generata un'enorme quantità di conclusioni irrilevanti.

Il ragionamento all'indietro inizia invece esaminando le conclusioni delle implicazioni. Occorre trovare una conclusione che corrisponda al problema da risolvere, e quindi passare ad esaminare le condizioni (se ce n'è almeno una) da cui dipende quella conclusione. Le condizioni vengono sottoposte al medesimo trattamento del problema originario (diventano le mete da raggiungere). Se tutte le condizioni alle quali la conclusione è subordinata, vengono dimostrate, si ottiene una soluzione del problema (una risposta alla domanda). In generale, il ragionamento all'indietro è più efficiente del ragionamento in avanti, in quanto compie solo deduzioni rilevanti per il problema da risolvere<sup>51</sup>.

---

<sup>50</sup>Questa strategia di risoluzione dei problemi da al PROLOG quel carattere procedurale sul quale ci siamo già soffermati nel cap. 1.

<sup>51</sup>La contrapposizione tra ragionamento in avanti e all'indietro (backward-forward reasoning) viene espressa usando numerose metafore, tratte da diversi settori dell'informatica: si parla di concatenamento in avanti-all'indietro (backward-forward chaining), elaborazione dall'alto in basso-dal basso in altro (bottom up-top down processing), inferenza guidata dai dati-guidata dal programma (data driven-problem driven inference) ecc. Su questa problematica, cfr. SHAPIRO S.C., Processing, Bottom-up and Top-down, in SHAPIRO S.C., ECKROTH D., Encyclopedia of Artificial Intelligence, cit. p. 779-785. Per una considerazione di tipo logico, cfr. KOWALSKI R.A., Logic for Problem Solving, cit., p. 53 ss. Possiamo forse dire (senza voler ridurre a questi schemi la ricchezza del ragionamento giuridico) che nel mondo del diritto, in linea di massima:

-il giudice ragiona all'indietro: data l'affermazione di un diritto (un effetto giuridico) proposta nell'azione, il problema è dimostrare se sussistano le condizioni alle quali l'ordinamento giuridico collega la nascita e la permanenza di quel diritto;

Il PROLOG, usa una strategia di ragionamento all'indietro. Per rispondere ad un quesito (soddisfare una meta) il sistema esamina il programma dall'inizio finché non trova una clausola la cui testa possa essere unificata con (cioè corrisponda a) il quesito. Due espressioni sono unificabili se le due espressioni possono essere rese uguali sostituendo uniformemente dei termini (variabili, costanti, termini composti) alle variabili che in esse compaiono. Ad esempio i due enunciati `uomo(x)` e `uomo(Socrate)`, possono essere unificati con la sostituzione

`x:=Socrate.`

che trasforma anche il primo enunciato nell'espressione `uomo(Socrate)`<sup>52</sup>.

I due enunciati

`uomo(Socrate)`  
`uomo(Platone)`

invece non sono unificabili.

La sostituzione unificatrice più generale di due enunciati, cioè quella che evita specificazioni non necessarie, viene chiamata unificazione di quegli enunciati.

Ad esempio, l'unificazione di

`uomo(x)` e  
`uomo(y)`,

potrà essere `y:=x`, che come risultato

`uomo(x)`,

non potrà essere invece `x=Socrate` e `y=Socrate` che dà

`uomo(Socrate)`,

poiché la sostituzione del nome di un individuo alle variabili è una concretizzazione (una specificazione) non necessaria al fine di rendere uguali le due espressioni.

Se la meta che il sistema sta cercando di soddisfare viene unificata con una clausola determinata, diciamo che quella clausola è attivata.

---

-l'avvocato ragiona in avanti: data la descrizione di certi fatti prospettata dal cliente, il problema è individuarne la rilevanza giuridica, i possibili effetti giuridici.

<sup>52</sup>In questa sede non approfondiamo formalmente il concetto di unificazione. Per una definizione formale, cfr. KOWALSKI R.A., *Logic for Problem Solving*, cit., p. 70.

Supponiamo di porre al nostro sistema (la cui base di conoscenza comprende gli enunciati riportati nella fig. 1 seguente) la domanda di verifica:

**<-genitore(Gino, Carlo) .**

In questo caso l'unificazione riesce quando il sistema trova il fatto **genitore(Gino, Carlo)**, che corrisponde esattamente alla predicazione usata nella domanda.

Esaminiamo ora un caso più complesso. Supponiamo di porre al sistema la domanda di ricerca.

**padre(Gino, y) .**

Viene attivata la regola

**padre(x, y) <- genitore(x, y) & maschio(y)**

L'unificazione della testa della regola con il quesito da luogo alla sostituzione **x:=Gino**,

A questo punto il sistema passa a considerare la coda della regola attivata (così come risulta dalla sostituzione), cioè la congiunzione

**<-genitore(Gino, y) & maschio(y)**

Queste enunciati diventano i sottoproblemi da risolvere (le nuove mete da soddisfare) l'uno dopo l'altro, da destra a sinistra.

La meta

**<-genitore(Gino, y)**

viene unificata con il fatto

**genitore(Gino, Carlo)**

La sostituzione **y:=Carlo** viene applicata anche all'atomo

**maschio(y)**

e da luogo alla meta

**<-maschio(Carlo)**

che coincide (può essere unificato con) il fatto identico presente nella base di dati.

In questo modo abbiamo soddisfatto tutte le condizioni poste dalla regola, e quindi il quesito originario.

La variabile **x**, nel corso del processo di unificazione è stata legata a (sostituita con) la costante Lucia e la risposta del sistema sarà infatti:

**y=Carlo.**

Se non è possibile soddisfare un quesito, il sistema torna adotta la strategia del c.d. ritorno indietro (*backtracking*): cerca di soddisfare il quesito usando clausole diverse da quelle sinora considerate<sup>53</sup>.

Ad esempio, se poniamo al nostro sistema la domanda

**madre(x, Carlo)**

Il sistema userà la regola

**madre(x,y) <- genitore(x,y) & femmina(y).**

La domanda e la testa della clausola possono essere unificate con la sostituzione

**y:=Carlo.**

Questa sostituzione trasforma il corpo della regola in:

**madre(x, Carlo) <- genitore(x, Carlo) & femmina(x).**

Si passa a risolvere il sottoproblema **genitore(x, Carlo)**. Come nel caso precedente, l'atomo **genitore(x, Carlo)**, viene unificato con il fatto **genitore(Gino, Carlo)**, ma la meta successiva, **femmina(Gino)** non può essere dimostrata (fallisce). Il sistema cercherà allora una nuova clausola con la quale unificare la domanda precedente **genitore(x, Carlo)**. Si tratterà del fatto **genitore(Amalia, Carlo)**. La meta successiva diventa **femmina(Amalia)**, che viene soddisfatta conducendo alla risposta

**x=Amalia.**

---

<sup>53</sup>Non approfondiamo l'esame della strategia inferenziale del Prolog. Anche su questo tema si rinvia ai manuali sul PROLOG.



**Fig. 1. Schema di un sistema PROLOG**

**Base di conoscenza (spazio di lavoro)**

```

maschio(Gino)
femmina(Amalia)
genitore(Gino, Carlo)
genitore(Amalia, Carlo)
coniuge(Giovanni, Lia)
italiano(x) <- genitore(y, x) & italiano(y)
italiano(x) <- coniuge(y, x) & italiano(y)
padre(x, y) <- genitore(x, y) & maschio(y)
madre(x, y) <- genitore(x, y) & femmina(y)
  
```

**Motore inferenziale del PROLOG**

**Interfaccia (input-output)**

**domande**

**risposte**

**utente**

### **3. Le regole di produzione**

Numerosi sistemi esperti rappresentano la conoscenza nella forma di regole, o come se dice anche, di regole di produzione.

Si tratta di enunciati ipotetici del tipo

"se <condizione> allora <conseguenza>" o  
 "<conseguenza> se <condizione>".

dove <condizione> è una condizione o un insieme di condizioni in presenza delle quali la regola risulta applicabile; <conseguenza> è la conseguenza (logica) che la regola prevede per la presenza di quelle condizioni.

In altri casi le regole assumono piuttosto la forma:

"se <condizione> allora <azione>"

che si ricollega, al modello comportamentista di stimolo e risposta, e dove <azione> è il comportamento che il sistema deve tenere quando si verificano le condizioni<sup>54</sup>.

In taluni sistemi, alle regole di produzione è associato un coefficiente probabilistico che indica l'affidabilità delle regole, cioè la probabilità che alle condizioni segua la conseguenza.

Ad esempio, un sistema esperto sviluppato presso la General Motors, per la localizzazione di guasti delle automobili, contiene regole del tipo:

#### Regola 408:

C è un'automobile.

Se : lo schema osservato attaccando un oscilloscopio al circuito di carico dell'automobile sono curve ad arco fluttuanti, e l'alternatore dell'automobile risponde correttamente a carichi differenti,

allora: vi è una forte probabilità (90%) che sia guasto il regolatore di voltaggio dell'automobile C.

Anche il celebre sistema MYCIN contiene regole probabilistiche, ad esempio:

se 'la colorazione dell'organismo è gram-positivo' e  
 'la morfologia dell'organismo è cocco' e  
 'il tipo di crescita dell'organismo è a gruppi',  
 allora 'l'identità dell'organismo è *strafilococco*': 60.

(il parametro 60 indica che possiamo dare un 60% di affidamento alla regola).

Nei sistemi che usano regole con conclusione probabile il motore inferenziale comprende un algoritmo numerico che determina l'affidabilità delle risposte del sistema sulla base dell'affidabilità delle regole impiegate nella loro deduzione.

<sup>54</sup>KOWALSKI R.A., *Logic Programming*, cit. p. 552 s..

Così, qualora tutte le condizioni della regola precedente siano state verificate (al 100%) il sistema può concludere che vi è un 60% di probabilità di essere di fronte ad un'infezione da stafilococchi. Un'altra regola di produzione, farà dipendere, ad esempio, la presenza di una polmonite, dalla presenza di un'infezione da streptococchi con determinate caratteristiche. Al fine di determinare la probabilità che vi sia una polmonite, il sistema dovrà rapportare il fattore di probabilità che vi sia un'infezione da streptococchi, al fattore di probabilità delle restanti condizioni della polmonite e al fattore di probabilità della regola usata.

Le regole di produzione sono il formalismo più usato nell'intelligenza artificiale e, in particolare, nelle applicazioni che hanno condotto a risultati operativi. La tecnologia dei sistemi basati su regole ha offerto per prima una metodologia e una notazione praticabili per lo sviluppo di sistemi di conoscenza<sup>55</sup>. Tra i vantaggi di questa tecnologia, (a) la facilità di rappresentare conoscenza pratica (possiamo esprimere come regole vari tipi di conoscenza: generalizzazioni empiriche, condizioni per raggiungere certi scopi, regole tecniche, regole prescrittive ecc.), (b) la possibilità di sviluppare agevolmente la base di conoscenza del sistema (aggiungendo nuove regole), (c) l'uso di tecniche inferenziali semplici ed efficienti, (d) la facilità di spiegare il comportamento del sistema (basta ripercorrere le regole applicate).

Il metodo delle regole di produzione (soprattutto quando si tratti di regole della logica enunciativa) si rivela invece insufficiente per rappresentare contesti complessi, che richiedano un complessivo modello concettuale del settore di cui trattasi.

Il sistemi che rappresentino la conoscenza mediante regole di produzione del tipo se <condizione> allora <conseguenza> e che adottino il ragionamento all'indietro sono di regola facilmente trasformabili in sistemi PROLOG. Infatti, le regole di produzione tendono a corrispondere alle clausole di Horn. Tuttavia, l'uso della programmazione logica spesso consente di ottenere una rappresentazione della conoscenza più semplice e generale<sup>56</sup>.

Ad esempio le due regole tipo EMICYN:

---

<sup>55</sup>Cfr. HAYES-ROTH F., *Rule-based Systems*, in SHAPIRO S.C., ECKROTH D., *Encyclopedia of Artificial Intelligence*, cit., p. 965.

<sup>56</sup>In PROLOG si possono realizzare sistemi che ragionano in avanti, grazie a tecniche di metaprogrammazione. Il comportamento dei sistemi che operano in base allo schema se <condizione> allora <azione> è invece difficilmente riproducibile in schemi logici (cfr. KOWALSKI R., *Logic programming*, cit., p. 553), e in generale, difficilmente controllabile.



I nodi indicano dei concetti o degli oggetti concreti (spesso è opportuno distinguere tra i due tipi di nodi), le frecce indicano rapporti tra le entità rappresentate dai nodi. Ad esempio, possiamo rappresentare così il fatto che il sig. Rossi lavora nel reparto produzione<sup>60</sup>:

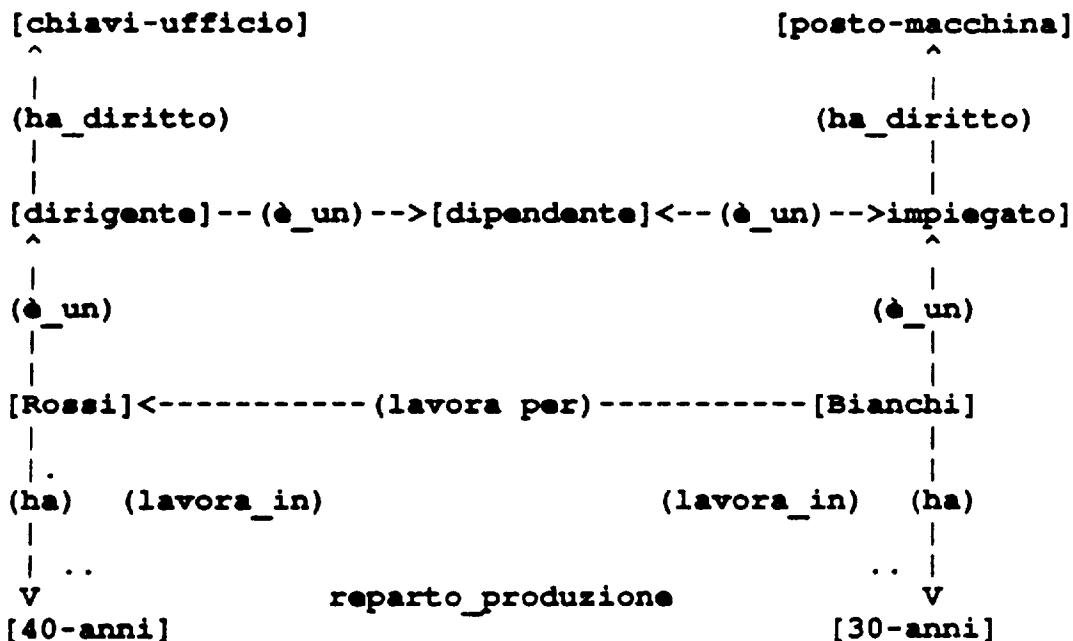
[Rossi]>----(lavora\_in)---->[reparto-produzione]

Anche le relazioni del tipo **è\_un** possono essere archi (frecce) di una rete semantica.

[Rossi]>----(è\_un)---->[dirigente]

Combinando rappresentazioni semplici come quelle appena illustrate possiamo ottenere reti semantiche complesse

**Fig. 2. Rete semantica**



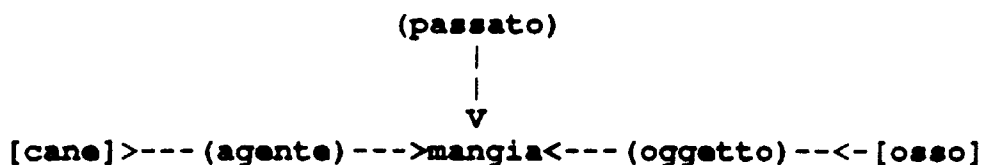
Si osservino le relazioni del tipo **è\_un**: grazie all'ereditarietà delle proprietà, possiamo inferire che Rossi ha diritto alle chiavi dell'ufficio, mentre sia Rossi che Bianchi hanno il posto macchina.

1961; CECCATO S., *Automatic Translation of Languages*, in "Inf. Stor. Retr.", 2 (3), 1964).

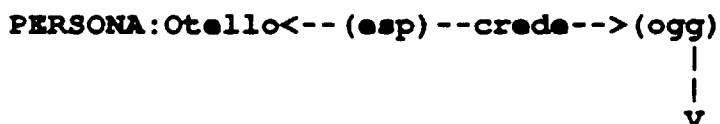
<sup>60</sup>Alcuni degli esempi presentati nelle pagine seguenti, sono stati tratti da ALTHY J.L., COOMBS M.J., *Expert Systems. Concepts and Examples*, The National Computing Centre, Manchester, 1984.

Sono stati sviluppati numerosi formalismi ispirati alle reti concettuali.

Il formalismo più semplice è quello dei grafi relazionali, che consistono di semplici nodi collegati da archi, come negli esempi presentati sopra. La capacità espressiva dei grafi relazionali è limitata ad una parte della logica predicativa<sup>61</sup>. Uno sviluppo dei grafi relazionali è rappresentato dai grafi incentrati sui verbi (*verb-centered graphs*) nei quali il verbo è considerato il punto focale di ogni proposizione. Il verbo regge le altre parti della frase (soggetto e complementi) e le proprietà dell'intera proposizione (ad esempio, il tempo) e relazioni con altre proposizioni vengono associate ad esso vengono collegate.



Un ulteriore sviluppo si ha con le reti proposizionali (nelle quali anche proposizioni possono essere nodi della rete) che consentono di rappresentare con maggior esattezza e facilità i rapporti tra proposizioni e di definire un contesto proposizionale che può fungere da argomento per altri operatori (ad esempio, per verbi modali come "dovere" e "potere", o quelli che esprimono atteggiamenti epistemici, come "pensare", "credere", "sapere" ecc.). Ad esempio, il seguente grafo corrisponde alla proposizione "Otello crede che Desdemona ami Cassio".



**PROPOSIZIONE:**

[PERSONA:[Desdemona]<--(esp)--<[ama]>--(ogg)>[PERSONA:Cassio]

Dove **esp** significa esperienza (chi prova una certa esperienza) **ogg** significa oggetto, **PERSONA** è il tipo cui appartiene l'individuo designato dal nome che segue i due punti.

E' possibile rappresentare le reti anche con notazioni lineari. Ad esempio, si possono usare parentesi quadre per

<sup>61</sup>Cfr. SOWA J.F., *Semantic Networks*, cit., p. 1111.

rappresentare i concetti e parentesi rotonde per rappresentare le relazioni concettuali. Il grafo della precedente potrebbe essere rappresentato come:

```
[PERSONA:Otello]<--(exp)<--[crede]<-->(ogg)<-->
-->[ PROPOSIZIONE:[PERSONA:Desdemona]<--
<--(esp)<--<[ama]>--(ogg)<-->[PERSONA:Cassio] ]
```

I sistemi basati sulle reti semantiche proposizionali possono avere una capacità espressiva che comprende quella della logica dei predicati.

Pertanto, alcuni autori hanno concepito lo sviluppo del linguaggio delle reti semantiche non come un'alternativa alla logica, ma piuttosto come un linguaggio logico alternativo rispetto alla notazione di Peano-Russel<sup>62</sup>, più vicino al linguaggio naturale, più facilmente estendibile a rappresentare anche una concezione prototipica del significato<sup>63</sup>, linguaggio che in taluni casi consentirebbe una maggiore efficienza computazionale<sup>64</sup>. Ad esempio i grafi concettuali di Sowa riprendono i grafi esistenziale di Pierce come notazione per la logica del primo ordine (con estensioni alla logica modale e alle logiche di ordine superiore), sulla quale basare il ragionamento deduttivo, ma consentono anche una rappresentazione di schemi e prototipi su cui fondare inferenze plausibili<sup>65</sup>.

I sistemi che usano reti semantiche includono, in generale, la possibilità di rappresentare tassonomie gerarchiche di tipi e sottotipi<sup>66</sup>.

Ad esempio, la fig. 3 seguente riporta una rete semantica che rappresenta una gerarchia di tipo è\_un.

---

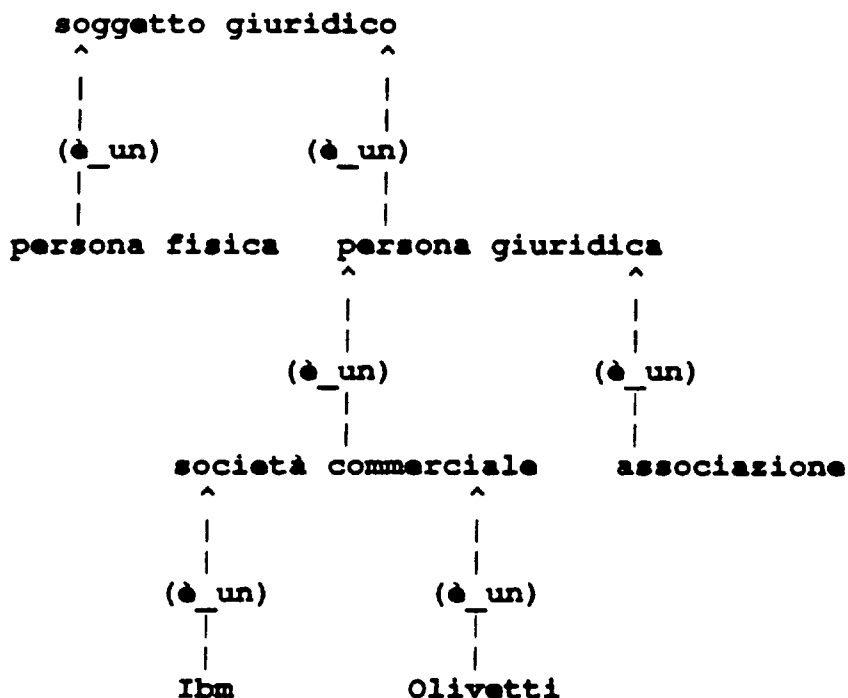
<sup>62</sup>La notazione logica usuale.

<sup>63</sup>Cfr, cap. 5 par. 2.1.

<sup>64</sup>Così SOWA J.F., *Semantic Networks*, cit., p. 1018.

<sup>65</sup>Così SOWA J.F., *Conceptual Structures*, cit., pp. 20-21.

<sup>66</sup>Seguendo il modello delle gerarchie di tipo è\_un sulle quali ci siamo soffermati in precedenza.

Fig. 3. Gerarchia di tipo è\_un (isa)

Anche le reti semantiche possono essere riformulate in PROLOG:

```

persona_fisica è_un soggetto_giuridico
individuo è_un persona_fisica
società_per_azioni è_un persona_giuridica
Ibm è_un società_per_azione
Olivetti è_un società_per_azioni

```

```

x è_un z <- x è_un y & y è_un z.

```

(regola che stabilisce la transitività della relazione è\_un<sup>67</sup>).

Si noti la fallacia connessa alla possibilità di trattare i tipi come individui. Possiamo aggiungere alla nostra rete la clausola:

```

persona_giuridica è_un categoria_giuridica

```

che significa "la persona giuridica è una categoria giuridica". Alla domanda "la Ibm è una categoria giuridica?", cioè

<sup>67</sup>Cfr. SOWA J.F., *Conceptual Structures*, cit., p. 19 s.



<-Ibm è un categoria\_giuridica

il sistema risponderà affermativamente<sup>68</sup>.

## 5. I frames

Spesso, nell'affrontare nuove realtà, non partiamo dal nulla, ma da strutture appropriate da riempire con i particolari delle esperienze concrete, usiamo prototipi o modelli ai quali rapportiamo la molteplicità dell'esperienza. Nell'intelligenza artificiale, per far fronte alle esigenze di rappresentare la realtà in schemi precostituiti, salvaguardando però le peculiarità dei singoli casi, si ricorre a schemi di vario tipo e in particolare ai c.d. *frames*<sup>69</sup>.

Un *frame* può essere visto come un formulario<sup>70</sup> con caselle (*slots*) da riempire con informazioni adeguate. Alcune caselle potranno essere riempite con dati arbitrari, altre solo con valori tratti da un insieme limitato, alcune possono avere un valore per difetto (di *default*), da adottarsi di regola, quando non sia diversamente specificato. I rapporti tra *frames* possono essere di tipo gerarchico (ad esempio del tipo *è un*). I *frames* di livello inferiore potranno ereditare le caratteristiche dei *frames* di livello superiore.

Ad esempio, nella figura seguente riportiamo un *frame* relativo al concetto di dirigente:

---

<sup>68</sup>Sul problema, cfr. SOWA J.F., *A Prolog to Prolog*, in WALKER A., MCCORD M., SOWA J.F., WILSON W.G., *Expert Systems and Prolog*, cit. , p. 46, e, più ampiamente, SOWA J.F., *Conceptual Structures*, cit. p. 84 s.

<sup>69</sup>La teoria dei *frames* nell'intelligenza artificiale ebbe origine dal tentativo di M. Minski di sviluppare uno schema nel quale rappresentare la conoscenza in modo strutturato, ma flessibile, ispirato al ragionamento comune. Cfr. MINSKI M., *A Framework for the Representation of Knowledge*, in WINSTON P. (a cura di), *The Psychology of Computer Vision*, McGraw-Hill, New York, 1975 1975, 211-280 (già pubblicato come *Artificial Intelligence Memo 306*, MIT AI Lab, 1974). Il nome *frame* che qui lasciamo in inglese, viene spesso tradotto in italiano come "cornice", ma dovrebbe forse tradursi più esattamente come "fotogramma".

<sup>70</sup> Infatti, i *frames* soddisfanno esigenze analoghe a quelle per le quali sono stati inventati i moduli e i formulari).

```

nome: DIRIGENTE
  è_un: DIPENDENTE
  nome: (nome, cognome)
  età: (anni)
  indirizzo: INDIRIZZO
  dipartimento: (produzione; amministrazione)
  salario: SALARIO
  data_inizio: (mese, anno)
  fino_a: (mese, anno) (default: (mese corrente, anno
                    corrente))

```

Il *frame* è identificato da un nome, e comprende una serie di elementi (le caselle) che rappresentano i componenti del concetto. La prima casella è del tipo *è un*, e ci dice che il concetto di dirigente è una specie del concetto di dipendente. Per talune delle caselle del *frame* sono definite delle restrizioni: se dopo l'etichetta della casella vi è un'espressione composta tra parentesi, il valore da inserire dovrà essere un dato composto nel modo specificato (ad esempio, *data\_inizio* si comporrà di un giorno e di un anno); se vi sono più espressioni separate da punto e virgola, il valore dovrà essere una di queste espressioni (nell'esempio, la casella *dipartimento* avrà per valore una delle due espressioni *produzione* e *amministrazione*). Si sono indicati in maiuscolo le caselle che debbono essere riempite non da dati semplici, ma da altre caselle.

Così la casella "indirizzo" può far riferimento al *frame*:

```

nome: INDIRIZZO
  città: (nome città)
  codice_postale: (numero)
  casa: (via, numero)

```

In questo modo, è possibile organizzare i concetti generali di un certo settore.

Gli individui vengono rappresentati riprendendo i *frames* di concetti generali e riempiendone le caselle con i dati relativi all'individuo (si parla di "istanziamento"). Così il *frame dirigente* può essere istanziato nel *frame Ross\_1* (attribuiamo a Rossi l'identificatore *Ros\_1*, in modo da distinguerlo da eventuali omonimi).

```

nome: Ros_1
  è_un: DIPENDENTE
  nome: Rossi, Mario
  età: 40
  indirizzo: ind_1
  dipartimento: produzione
  salario: sal_1
  data_inizio: (aprile, 1980)
  fino_a: (mese_corrente, anno_corrente)

```

Ci siamo limitati a illustrare *frames* che rappresentano oggetti e situazioni statiche. Sono stati proposti anche tipi di *frame* (i c.d. copioni (*script*)) per rappresentare sequenze di eventi che si svolgono contesti determinati<sup>71</sup>.

Tra i vantaggi dell'uso dei *frames*, si possono indicare la flessibilità, la possibilità di rappresentare informazioni con struttura complessa, la capacità di incorporare aspetti procedurali; tra i limiti, la mancanza di una semantica e di metodi di inferenza definiti con rigore logico. Recentemente, tuttavia, sono stati sviluppati sistemi ibridi, nei quali un linguaggio di tipo *frames* è usato per rappresentare definizioni e rapporti concettuali, mentre si adotta un diverso linguaggio (di solito ispirato alla programmazione logica) per la rappresentazione delle asserzioni che usano i concetti introdotti con strutture di *frames*. Vi sono inoltre interessanti tentativi di usare i *frames* come una rappresentazione di alto livello che possa essere trasformata automaticamente in una rappresentazione di tipo logico. Anche *frames* possono essere trascritti in PROLOG<sup>72</sup>.

---

<sup>71</sup>Cfr. SCHANK R., ABELSON R.P., *Scripts, Plans, Goals and Understanding*, Erlbaum, Hillsdale (NJ), 1977.

<sup>72</sup>Si tratta di un'operazione troppo complessa perché possa essere trattata in questa sede.



## CAP. 8

### METODI E LINGUAGGI

### PER LA RAPPRESENTAZIONE

### DELLA CONOSCENZA GIURIDICA

L'informatica giuridica ha preso a prestito le tecniche per la rappresentazione della conoscenza sviluppate in altri contesti, logici, matematici e informatici. Come si è osservato in precedenza, una rappresentazione di tipo algoritmico (anche secondo le prescrizioni della programmazione strutturata) incontra precisi limiti nel diritto. Di conseguenza, le ricerche in questa direzione non hanno condotto a risultati significativi e sono state sostanzialmente abbandonate negli anni 80. Nelle pagine seguenti ci limiteremo ad esaminare le applicazioni giuridiche della logica e di altri metodi per la rappresentazione della conoscenza adottati nell'intelligenza artificiale.

#### 1. La logica nel diritto (cenni)

Il tentativo di applicare la logica moderna al diritto, sebbene non vada al di là del nostro secolo, ha preceduto gli studi informatico-giuridici, e si è sviluppato indipendentemente da essi. Gli studi in materia di logica giuridica rispondono a due esigenze fondamentali:

(a) l'esigenza pratica di accrescere la chiarezza del linguaggio giuridico e il rigore del ragionamento dei giuristi;

(b) l'interesse teorico per la logica quale strumento di analisi del discorso giuridico.

I risultati pre-informatici dell'uso della logica nel diritto sono stati assai scarsi per quanto riguarda il punto (a), notevoli invece per quanto riguarda il punto (b). La logica non è penetrata minimamente nella formazione dei giuristi e nella pratica del diritto, mentre vi è ormai una tradizione consolidata nell'uso di strumenti logico-formali per lo studio della struttura dell'ordinamento giuridico e per l'analisi del ragionamento giuridico.

Lo scarso impatto della logica (e, in generale, dei metodi formali) sulla vita del diritto attiene, dipende dal fatto che il ragionamento naturale di regola è sufficiente

per il controllo delle inferenze deduttive dei giuristi<sup>1</sup>. Ciò non significa, tuttavia che si debba ricorrere alla logica per la rappresentazione e l'uso della conoscenza giuridica in un sistema informatico-giuridico intelligente. Infatti, un sistema automatico, proprio perché manca del ragionamento naturale, deve ricorrere a strumenti formali. Grazie a questi strumenti si realizzerà non un giurista automatico, ma piuttosto un nuovo interessante strumento di lavoro per il giurista, nell'ambito e nei limiti delle tecnologie dell'intelligenza artificiale. Come si è già osservato<sup>2</sup> tali tecnologie possono essere usate per la deduzione delle conseguenze logiche di un modello assiomatico del diritto, e in misura molto limitata, ricorrendo a metodi euristici, anche per la costruzione di un modello siffatto. L'uso di un modello assiomatico del diritto può essere utile non solo nei settori dove non si pongono gravi problemi interpretativi (per operazioni di routine), ma anche al fine di garantire la controllabilità e l'universalizzabilità della decisione giuridica (a condizione che sia garantita la possibilità di un continuo riesame critico della base assiomatica).

Numerosi filosofi e teorici del diritto si sono soffermati sul rapporto tra logica e diritto. Purtroppo, anziché analizzare problemi concreti e proporre soluzioni determinate<sup>3</sup> ci si è spesso limitati all'annoso dibattito sull'astratta "applicabilità della logica al diritto", dibattito privo di qualsiasi implicazione significativa, pratica o teorica, per l'informatica giuridica (e, forse, anche per la filosofia del diritto).

Le obiezioni sollevate contro l'applicabilità della logica al diritto possono essere ricondotte a due problematiche distinte:

(a) Una concezione prescrittivistica delle norme (e talvolta anche degli enunciati della scienza giuridica) ha condotto ad escludere che alle norme (e agli enunciati della scienza giuridica) sia applicabile la logica del vero e del falso. Quindi, questa concezione ha condotto ad escludere ogni impiego di tecniche logiche in contesti etico-giuridici, oppure ha condotto a proporre logiche concernenti anche o solo enunciati non descrittivi. Sotto questo primo profilo, la questione del rapporto tra logica e diritto si inquadra nella problematica logico-filosofica dello statuto

---

<sup>1</sup>Ci siamo soffermati brevemente su questo tema nel cap. 6.

<sup>2</sup>Cfr. cap. 6 par. 3.

<sup>3</sup>Ad esempio, come si è rilevato, manca ancora una logica deontica soddisfacente.

(b) L'impiego delle tecniche della logica moderna (la cd. logica formale) è apparso strettamente connesso con un approccio formalistico ai problemi giuridici. Quindi, la discussione del rapporto tra logica e diritto si è situata nell'ambito della problematica del metodo giuridico e, in particolare, nell'ambito del secolare dibattito tra formalismo ed antiformalismo giuridici.

Si tratta di due questioni da separare nettamente tra loro. La prima attiene principalmente alla sintassi e alla semantica logica: un'applicazione della logica al diritto è legittima sotto il profilo della sintassi e della semantica, è compatibile con i concetti di diritto e logica dai quali muoviamo?<sup>4</sup>.

La seconda questione attiene invece alla pragmatica, concerne non tanto la legittimità di un'applicazione della logica al diritto, quanto l'opportunità o l'utilità di una tale applicazione, in relazione a diverse concezioni del diritto e dell'attività del giurista<sup>5</sup>.

Gli studi informatico-giuridici sembrano poter incidere profondamente sugli studi logico-giuridici, in quanto pongono una gamma di problemi nuovi, diversi da quelli tradizionali, più concreti ma anche più vicini ai grandi temi della logica filosofica. Si pensi ad esempio, a problemi come la rappresentazione logica degli eventi, del tempo, lo sviluppo di logiche non monotoniche, l'efficienza computazionale delle logiche deontiche, il rapporto tra logica e strutture del linguaggio naturale, il rapporto tra ragionamento deduttivo e plausibile ecc. Il trattamento logico del linguaggio giuridico pone problemi di estrema difficoltà. Come si è già rilevato, il linguaggio giuridico è immerso nel linguaggio naturale, e inoltre il mondo del diritto comprende tutto il mondo sociale, non può essere ridotto a contesti predefiniti e semplificati (come nelle scienze esatte e naturali). Pertanto, una logica per l'informatica giuridica deve affrontare tutti i concetti del senso comune (spazio, tempo, modalità ecc.) cioè tutti i problemi della logica filosofica, senza dimenticare gli aspetti computazionali.

Nelle pagine seguenti, si illustreranno alcuni dei tentativi più interessanti di usare la logica per la rappresentazione formale della conoscenza giuridica, senza

---

<sup>4</sup>Su questi temi mi permetto di rinviare a SARTOR G., *Alcune osservazioni sull'applicabilità della logica classica al diritto*, in "Informatica e diritto", 3, 1987, pp. 65-90.

<sup>5</sup>Per quanto attiene agli aspetti pragmatici, in questa sede basta ricordare la distinzione già illustrata tra ragionamento umano "naturale" e ragionamento di un sistema automatico, che deve ricorrere necessariamente a strumenti formali.

approfondire gli aspetti formali al di là del minimo indispensabile.

### 1.1. La logica enunciativa. La normalizzazione di L.E. Allen

Sono stati sviluppati numerosi sistemi informatico-giuridici la cui base di conoscenza è espressa in formalismi ispirati alla logica enunciativa. In particolare, come abbiamo visto, si tratta di sistemi basati su regole di produzione (proposizioni ipotetiche) della logica enunciativa<sup>6</sup>.

Mi limiterò ad illustrare l'approccio proposto da L.E. Allen, l'autore che più ha approfondito il problema della rappresentazione della conoscenza giuridica nella logica enunciativa. Allen ha elaborato la *normalizzazione*, una tecnica che comprende (a) una metodologia per passare dal linguaggio naturale ad un linguaggio logico, e (b) un linguaggio (riconducibile alla logica enunciativa) che consente una rappresentazione logica di contesti giuridici<sup>7</sup>.

La tecnica della normalizzazione fu ideata da L.E. Allen quale strumento per migliorare la stesura dei testi giuridici, eliminandone le ambiguità sintattiche<sup>8</sup>. Allen chiama "normalizzazione" la procedura che conduce ad una riformulazione di un testo giuridico priva di talune ambiguità sintattiche (versione normalizzata). La versione normalizzata di un testo normativo è una sequenza di disposizioni normalizzate. In ciascuna disposizione normalizzata la connessione tra gli enunciati è espressa da connettivi logici, e gli argomenti di tali connettivi sono identificabili univocamente sulla base di convenzioni redazionali. Per esprimere le disposizioni normalizzate Allen ha proposto diverse sintassi equivalenti:

-*formule logiche*. Le formule logiche sono formule della logica enunciativa, nelle quali gli enunciati atomici sono rappresentati ciascuno da un breve identificatore (o etichetta). I connettivi logici usati da Allen comprendono i comuni connettivi della logica enunciativa (negazione,

---

<sup>6</sup>Ad esempio, il sistema Judith, il sistema sviluppato nel progetto CCLIPS, il sistema LSD, il sistema LABEO e molti altri dei sistemi illustrati nel cap. 4.

<sup>7</sup>La problematica della normalizzazione sarà ripresa più ampiamente nell'appendice, dove si illustrerà il sistema IRINORM, che rappresenta, in larga parte, uno sviluppo della normalizzazione di L.A. Allen

<sup>8</sup>Sulle prescrizioni di una metodologia per passare dal testo in linguaggio naturale, cfr. ALLEN L.E., *Una guida per redattori giuridici di testi normalizzati*, cit., in particolare p. 111 ss.



coniunzione, disgiunzione, condizionale, doppio condizionale), e inoltre altri connettivi, introdotti per esprimere più direttamente alcune strutture sintattiche del linguaggio naturale<sup>9</sup>;

-*diagrammi a freccia (arrow diagrams)*. I diagrammi a freccia rappresentano in forma grafica i rapporti logici tra gli identificatori degli enunciati atomici;

-*testi normalizzati (normalized texts)*. I testi normalizzati sono testi in linguaggio pseudonaturale, nei quali gli enunciati atomici sono riportati per intero, i connettivi logici sono formulati con espressioni più vicine al linguaggio naturale (nella figura seguente riportiamo per ciascun connettivo il simbolo logico e l'espressione più estesa corrispondente), e ogni connettivo ha per argomento gli enunciati allineati con esso. Per distinguere i connettivi logici dalle corrispondenti espressioni del linguaggio naturale, viene adottata una grafia particolare (essi vengono scritti tutti in maiuscolo, oppure con la lettera iniziale o finale maiuscola).

---

<sup>9</sup>Ad esempio, i connettivi **as (ANCHE SE)**, **sc (SALVO CHE)**, **ma (MA ALTRIMENTI)**. Nella figura seguente si riportano i principali connettivi proposti da Allen, e il significato dei nuovi connettivi viene specificato nei termini dei connettivi logici tradizionali. I termini sintattici sono riportati nella traduzione italiana adottata nel nostro programma IRINORM.

Fig. 1. I connettivi della normalizzazione

simbolo	espressione logica definita
n	NON SI VERIFICA CHE (negazione)
& E (congiunzione)	
v	O (disgiunzione debole: latino vel) $P \vee Q =df \neg (n P \& n Q)$
->	SE-ALLORA (implicazione) $P \rightarrow Q =df (n P) \vee Q$
<-	SE $P \leftarrow Q =df Q \rightarrow P$
eq	SE E SOLO SE (doppia implicazione) $P \text{ eq } Q =df (Q \rightarrow P) \& (n Q \rightarrow n P)$
aut	AUT (disgiunzione forte: latino aut) $P \text{ aut } Q =df (P \vee Q) \& n (P \& Q)$
sc	SALVO CHE $P \text{ sc } Q =df Q \rightarrow n P$
sc1	SALVO CHE1 $P \text{ sc1 } Q =df (Q \rightarrow n P) \& (n Q \rightarrow P)$
eq	SE E SOLO SE $P \text{ eq } Q =df (Q \rightarrow P) \& (n Q \rightarrow n P)$
ma	MA ALTRIMENTI $(P \rightarrow Q) \text{ ma } R =df (P \rightarrow Q) \& (n P \rightarrow R)$
as	ANCHE SE $(P \text{ as } Q) =df (Q \rightarrow P) \& ((n Q) \rightarrow P)$
O	E' OBBLIGATORIO CHE
P	E' PERMESSO CHE

1.1.1 Primo esempio : la normalizzazione della normativa sulla capacità di contrarre matrimonio

Allen e Saxon<sup>10</sup> hanno redatto la seguente versione normalizzata di alcuni articoli del codice civile italiano in materia di matrimonio<sup>11</sup>. La normativa è stata formalizzata (a) in una serie di disposizioni che collegano a certe condizioni la presenza di un requisito per il matrimonio e (b) in una regola che connette il potere di contrarre matrimonio alla presenza di tutti i requisiti (ciascuno necessario e, nel loro insieme sufficienti). Riportiamo dapprima una delle disposizioni specifiche e quindi la norma che raccoglie i requisiti.

a) Il requisito dell'età

**SE**

1. la persona è maggiorenne, O
2. A. La persona ha raggiunto il sedicesimo compleanno  
E,  
B. il giudice emette un decreto che consente il matrimonio di persone che abbiano raggiunto il sedicesimo anno

**ALLORA**

3. la persona soddisfa il requisito dell'età.

---

<sup>10</sup>ALLEN L.A., SAXON C.S., *Analysis of the Logical Structure of Legal Rules by a Modernized and Formalized Version of Hohfeld's Fundamental Legal Conceptions*, cit., p. 439 s.

<sup>11</sup>Si tratta di un testo che è stato oggetto di diversi esperimenti di formalizzazione presso l'Istituto di documentazione giuridica del Consiglio nazionale delle ricerche di Firenze. In particolare è stato usato anche un metodo algebrico proposto da Sanchez-Mazas (cfr. BIAGIOLI C., BIAGINI P., DINI G., MARTINO A.A., SOCCI F., TISCORNIA D., TRIVISONNO G., *An Arithmetic Model for Analysing Normative Systems: Some Experimental Proposals*, Istituto per la documentazione giuridica del Consiglio nazionale delle ricerche, Firenze, giugno 1982; SANCHEZ-MAZAS M., *Algebraic and Arithmetical Translations on Normative Systems and Applications in Legal Informatics*, in MARTINO A.A. (a cura di) *Deontic Logic, Computational Linguistics and Legal Information Systems*, cit., pp. 169-201.

b) Le condizioni del potere di contrarre matrimonio

SE

1. la persona soddisfa il requisito d'età per il matrimonio, E
2. la persona soddisfa il requisito di capacità mentale per il matrimonio, E
3. sono soddisfatti il requisito delle circostanze appropriate per il matrimonio, E,
4. la persona soddisfa il requisito di stato per il matrimonio, E
5. la persona soddisfa il requisito delle relazioni di parentela per il matrimonio, E,
6. la persona soddisfa il requisito della relazione mediante matrimonio, E,
7. la persona soddisfa il requisito della relazione adottiva per il matrimonio, E,
8. la persona soddisfa il requisito della proibizione temporanea per il matrimonio,

ALLORA

9. quella persona ha il POTERE di contrarre matrimonio  
MA ALTRIMENTI, NO.

1.1.2 Secondo esempio, la normalizzazione di un articolo della costituzione italiana

In questo secondo esempio, si illustrano i diversi modi nei quali può essere espresso un testo normalizzato. Si considererà il primo comma dell'art. 18 della Costituzione italiana.

a) Testo originale

I cittadini hanno diritto di associarsi liberamente, senza autorizzazione, per fini che non sono vietati ai singoli dalla legge penale<sup>12</sup>.

b) Formula logica

$$((c1 \ \& \ n-(c2 \ \& \ c3 \ \& \ c4)) \ > \ (as \ n-c5 \ > \ c9)) \ sc \ (c6 \ v \ (c7 \ \& \ c8))$$


---

<sup>12</sup>Si è dovuta inserire nel primo comma l'eccezione stabilita dal secondo comma che recita:

"Sono proibite le associazioni segrete e quelle che perseguono, anche indirettamente, scopi politici mediante organizzazioni di carattere militare.

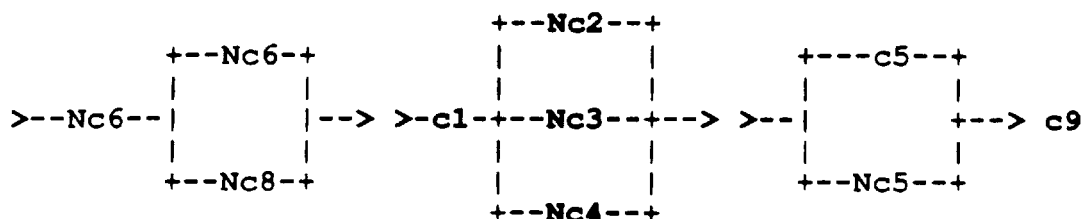
c) Testo normalizzato

1. SE
  - A. 1. [c1: l'associazione e' composta di cittadini] E
    2. NON SI VERIFICA CHE
      - A. 1. [c2: vige un divieto dello scopo  
perseguito dall'associazione] E
      2. [c3: il divieto e' posto dalla legge  
penale] E
      3. [c4: il divieto vale per i singoli]
  - ALLORA
  - B. ANCHE SE
    1. A. NON SI VERIFICA CHE
      1. [c5: l'associazione e' stata autorizzata]
    - ALLORA
    2. [c9: l'associazione e' permessa]
    - SALVO CHE
2. A. [c6: l'associazione e' segreta] O
  - B. 1. [c7: l'associazione persegue scopi politici] E
    2. [c8: l'associazione ha una organizzazione di  
tipo militare]

d) Schema del testo normalizzato

1. SE
  - A. 1. c1 E
    2. NON SI VERIFICA CHE
      - A. 1. c2 E
      2. c3 E
      3. c4
  - ALLORA
  - B. ANCHE SE
    1. A. NON SI VERIFICA CHE
      1. c5
    - ALLORA
    2. c9
    - SALVO CHE
2. A. c6 O
  - B. 1. c7 E
    2. c8

### Diagramma a freccia



Seguendo un percorso che conduca dall'estrema destra all'estrema sinistra del diagramma si può ottenere l'effetto giuridico c9 (l'associazione è permessa)<sup>13</sup>.

#### 1.1.3 Il significato della normalizzazione

Grazie alla normalizzazione, si può ottenere un testo al tempo stesso facilmente comprensibile dall'uomo (in quanto apparentemente vicino al linguaggio naturale) e dotato di una struttura logicamente definita. Allen, come abbiamo visto, ha sviluppato un programma (NORMALIZER)<sup>14</sup> che consente di trasformare un'espressione logica in un testo normalizzato<sup>15</sup>. Tale trasformazione è particolarmente semplice in quanto il testo prodotto automaticamente è espresso in linguaggio naturale solo in apparenza. In realtà si tratta di un testo dotato di una struttura logicamente definita: la normalizzazione è solo la proposta (a) di una diversa sintassi nella quale esprimere formule logiche, e (b) di una metodologia per passare da un testo in linguaggio naturale ad espressioni logicamente definite.

Questa sintassi presenta alcuni aspetti interessanti, e consente l'uso di strumenti informatici nella redazione di testi giuridici. Strumenti come NORMALIZER possono facilitare la redazione di testi giuridici in un linguaggio parzialmente formalizzato, poiché consentono di evitare il

<sup>13</sup>Nel diagramma a freccia abbiamo semplificato le strutture sintattiche usate nelle rappresentazioni precedenti. Il diagramma corrisponde direttamente alla formula logica

$n-c6 \ \& \ (n-c7 \vee n-c8) \rightarrow ((c1 \ \& \ n-(c2 \ \& \ c3 \ \& \ c4)) \ > \ ((n-c5 \vee c5) \ > c9))$

equivalente alla precedente.

Si tratta della versione che Allen chiama chiara, che contiene cioè solo i connettivi n-, & e -> (cfr. appendice, par. 3.1.1).

<sup>14</sup>Cfr. cap. 4 par. 2.6.

<sup>15</sup>Non ci diffondiamo su questo passaggio, che esamineremo nell'appendice, nel par. 3.1.1.

lavoro manuale necessario per passare dalle espressioni sintetiche delle formule logiche ad una rappresentazione più estesa e leggibile.

NORMALIZER consente di ottenere più versioni logicamente equivalenti del medesimo testo normalizzato (sfruttando le equivalenze logiche)<sup>16</sup>. In questo modo, l'utente può redigere testi giuridici scegliere la redazione più sintetica e leggibile, e controllare l'esattezza della formalizzazione usando le strutture logiche più semplici. Allen muove dall'ipotesi che tutte le norme siano esprimibili come implicazioni, ipotesi che si traduce in regole sintattiche per la redazione dei testi normalizzati. Pertanto i testi normalizzati possono essere trasformati mediante equivalenze logiche in congiunzioni di implicazioni (o, se si preferisce, in regole di produzione)<sup>17</sup>.

La tecnica della normalizzazione, così come elaborata da Allen, tocca solamente la struttura enunciativa del testo: la formalizzazione è limitata ai rapporti tra enunciati (e, più esattamente, ai soli rapporti tra enunciati esprimibili mediante i connettivi logici). Ciò comporta significative limitazioni: l'ambiguità sintattica all'interno degli enunciati non è disambiguabile nel testo normalizzato; è impossibile esprimere in modo adeguato il fatto che più enunciati fanno riferimento al medesimo oggetto senza ripetere il nome di tale oggetto in ogni enunciato (così per tutti i riferimenti impliciti nell'enunciato di cui si tratta e esplicitati solo negli enunciati contigui); non si possono formalizzare le norme generali<sup>18</sup>.

Inoltre, la normalizzazione è solo una sintassi nella quale esprimere formule logiche, non è ancora una logica per rappresentare la conoscenza giuridica. A tal fine bisogna associare alla normalizzazione una logica determinata. A questo proposito, Allen e Saxon hanno proposto una complessa

---

<sup>16</sup>Anche questo aspetto sarà trattato più ampiamente nell'appendice, par. 3.1.1.

<sup>17</sup>Questa trasformazione non sempre risulta possibile in base alla definizione di testo normalizzato proposta in ALLEN. L.A., *Towards a Normalized Language to Clarify the Structure of Legal Discourse*, cit., pp. 352 ss., nell'ambito della logica enunciativa presentata in ALLEN L.A., SAXON C.S., *Analysis of the Logical Structure of Legal Rules by a Modernized and Formalized Version of Hohfeld's Fundamental Legal Conceptions*, cit.

<sup>18</sup>Nel programma IRINORM si è tentato di realizzare una normalizzazione al livello della logica dei predicati.

"logica neohohfeldiana", che riprende, la tradizione delle logiche rilevanti<sup>19</sup>. E' da osservare a riguardo che

- (a) la logica neohohfeldiana è estremamente complessa;
- (b) la mancanza di una semantica (aspetto non sviluppato da Allen e Saxon) fa sì che la definizione di ogni connettivo dipenda solo da regole di inferenza, difficili da padroneggiare e da apprezzare nel loro complesso;
- (c) l'opportunità dell'uso delle logiche rilevanti nel diritto suscita talune perplessità;
- (d) non sono stati considerati, nello sviluppo di tale sistema logico, i problemi dell'efficienza computazionale.

Per altri aspetti, tuttavia, e in particolare per lo sviluppo dei concetti deontici e il loro rapporto con i concetti modali, la logica neohohfeldiana presenta un estremo interesse.

Recentemente Allen e Saxon hanno sviluppato il sistema AUTOPROLOG, che crea una base di conoscenza in PROLOG, a partire da un testo normalizzato, e consente di interrogare la base di conoscenza<sup>20</sup>. Tale sistema usa le capacità inferenziali del PROLOG solo in limiti assai ristretti. Infatti il testo è formalizzato nella logica enunciativa, e il sistema, pertanto, si limita alle inferenze della logica enunciativa e ogni enunciato come un predicato con 0 argomenti<sup>21</sup>.

Si può pertanto concludere che, in quanto metodo per la redazione di testi giuridici in un linguaggio formalizzato, la tecnica della normalizzazione sembra di notevole efficacia. Invece, la logica che Allen ha sinora associato alla formalizzazione lascia numerosi problemi irrisolti, al fine della realizzazione di un sistema informatico-giuridico intelligente:

- (a) la logica enunciativa, la sola che Allen abbia usato finora nelle implementazioni informatiche, consente solo una

---

<sup>19</sup> Cfr. ALLEN L.A., SAXON C.S., *Analysis of the Logical Structure of Legal Rules by a Modernized and Formalized Version of Hohfeld's Fundamental Legal Conceptions*, cit..

<sup>20</sup> Cfr. ALLEN L.E., SAXON C.S., *Automatic generation of a legal expert system of section 7 (2) of the United Kingdom data protection act 1984*, cit.; ALLEN L.E., SAXON C.S., PAYTON S.A., *Syntesizing related Rules from Statutes and Cases for Legal Expert Systems*, cit.

<sup>21</sup> Il che significa che non sono possibili quesiti di ricerca, e che anche i quesiti di verifica sono possibili solo in misura estremamente limitata (non sarà possibile ricondurre il predicato specifico all'affermazione generale).



rappresentazione superficiale della conoscenza giuridica e, quindi, non permette un trattamento informatico adeguato<sup>22</sup>;

(b) la logica neohofeldiana non è stata ancora collegata alla normalizzazione, né in implementazioni informatiche, né grazie ad un completamento del linguaggio in cui esprimere i testi normalizzati che consenta di esprimere una formalizzazione di livello sub-enunciativo.

## 1.2. La logica predicativa

Una trattazione dell'uso della logica predicativa per la rappresentazione della conoscenza giuridica può trovarsi in alcuni manuali di logica per giuristi<sup>23</sup>. Mi limito a riportare due esempi di formalizzazione di contesti giuridici nella logica predicativa. Il primo concerne un principio generale del diritto civile, il secondo l'argomentazione di un avvocato.

### 1.2.1 Formalizzazione di un principio generale del diritto

Rödig propone la seguente formalizzazione di un principio logico-giuridico, (ma forse ancor prima, un principio del senso comune), non espresso nella legislazione, ma al quale i giuristi fanno riferimento nella ricostruzione dei casi che gli vengono sottoposti<sup>24</sup>.

#### a) Principio

"Una persona  $p^1$  (attore) può richiedere da un'altra persona (convenuto) la consegna di una cosa in un momento  $z^3$ , se e solo se, in primo luogo c'è un momento  $z^1$ , che non è successivo (in ogni caso) a  $z^3$ , nel quale l'attore ha ottenuto il diritto alla consegna, e, in secondo luogo, non c'è nessun momento  $z^2$  tra  $z^1$  e  $z^3$  (o contemporaneo a  $z^1$  o  $z^3$ ) nel quale l'attore ha perduto questo diritto"

---

<sup>22</sup>Ad esempio, un sistema automatico non potrebbe dedurre dai testi normalizzati che abbiamo riportato nelle figure precedenti, che Giorgio, che ha 15 anni non può sposare Laura, o che Massimo, che ha 35 anni e soddisfa tutti i requisiti, ulteriori rispetto all'età, la può sposare.

<sup>23</sup>Numerosi esempi di formalizzazione nella logica predicativa possono trovarsi, in particolare, in TAMMELO I., *Modern Logic in the Service of Law*, cit.

<sup>24</sup>Cfr. RÖDIG J., *Axiomatisierbarkeit juristischer Systeme*, in *Schriften zur juristischen Logik*, 1980, pp. 86 ss.

b) Glossario:

$\text{Res}[(1), (2), (3), (4)]$ : (1) può richiedere da (2) al momento (3) la restituzione di (4).  
 $\text{Ott}[(1), (2), (3), (4)]$ : (1) ottiene al momento (2) nei confronti di (3) il diritto alla restituzione di (4).  
 $\text{Per}[(1), (2), (3), (4)]$ : (1) perde al momento (2) nei confronti di (3) il diritto alla restituzione di (4).

$\text{Pr}[(1), (2)]$ : (1) è precedente a (2)<sup>25</sup>

c) Formalizzazione nella logica predicativa

$$\begin{aligned} (p^1) \ (p^2) \ (z^3) \ (s) \ (\text{Res}(p^1, p^2, z^3, s) \leftrightarrow) \\ \leftrightarrow \exists(z^1) \ (\neg \text{Pr}(z^3, z^1) \ \& \ \text{Ott}(p^1, p^2, z^1, s) \ \& \\ \& \neg \exists(z^2) \ (\neg \text{Pr}(z^2, z^1) \ \& \ \text{Pr}(z^2, z^3) \ \& \\ \& \ \text{Per}(z^1, p^2, z^2, s)). \end{aligned}$$

1.2.2 Formalizzazione di un'argomentazione giudiziaria

Come esempio della rappresentazione di un'argomentazione giuridica nella logica predicativa, si considererà l'argomentazione dalla difesa nella causa *Broom v. Morgan* (1953), così come viene sviluppata da Tammelo<sup>26</sup>.

a) Fatti della causa:

Un marito e sua moglie erano dipendenti nella stessa azienda. Per negligenza il marito ferì la moglie. La moglie citò in giudizio il datore di lavoro.

b) Testo dell'argomento della difesa:

La responsabilità vicaria e la responsabilità iniziale (diretta) sono due settori diversi della responsabilità. Se vi è solo responsabilità vicaria, una pre-condizione è che vi sia responsabilità da parte del soggetto agente - in questo caso il marito - che deve essere stato, nei confronti della moglie, nell'obbligo di non ferirla negligenemente.

<sup>25</sup>Le cifre tra parentesi indicano gli argomenti del predicato.

<sup>26</sup>Cfr. TAMMELLO I., *Modern Logic in the Service of Law*, cit. p. 97 ss.

Un marito non ha alcun dovere, imposto e riconosciuto dalla legge, di non causare lesioni alla moglie per negligenza, Anche se egli l'ha colpita alla testa, la moglie non ha alcun rimedio di diritto civile contro di lui, benché potrebbe querelarlo per aggressione. Non essendoci alcun tipo di azione riconosciuta dal diritto, non c'è illecito. "Illecito" significa un fatto dannoso (*wrong*) azionabile. Ciò che il marito ha fatto alla moglie non era un fatto dannoso azionabile e pertanto non si è verificato alcunché per cui l'imprenditore sia responsabile.

c) Struttura logica dell'argomento:

Nell'argomento la conclusione che il datore di lavoro non è responsabile viene dedotta da tre premesse:

1. Per ogni *x*, *x* commette un fatto illecito solo se *x* commette un danno azionabile.
2. Il marito non commette un fatto dannoso azionabile.
3. Se il marito non è responsabile allora il datore di lavoro non è responsabile.

Tammelo rileva che vi è una premessa ulteriore implicita, necessaria alla validità dell'inferenza (e pertanto il ragionamento è entimematico):

4. Una persona è responsabile se e solo se commette un illecito civile:

d) Glossario:

*F*(1): (1) commette un fatto illecito  
*D*(1): (1) commette un fatto dannoso azionabile  
*R*(1) : (1) è responsabile  
*mar*: il marito  
*dat*: il datore di lavoro"

e) Premesse dell'argomento:

- (1)  $\forall (x) ( F(x) \rightarrow D(x) )$

(se una persona commette un fatto illecito allora essa commette un danno azionabile).

- (2)  $\neg D(\text{mar})$

(il marito non ha commesso un danno azionabile).

(3)  $n\text{-}R(\text{mar}) \rightarrow n\text{-}R(\text{dat})$

(se il marito non è responsabile allora il datore di lavoro non è responsabile).

(4)  $R(x) \rightarrow F(x)$  (premessa implicita)

(una persona è responsabile solo se commette un fatto illecito).

#### f) Conclusione

(5)  $n\text{-}R(\text{dat})$

(il datore di lavoro non è responsabile).

#### g) Deduzione

	$V(x) \quad (F(x) \rightarrow D(x))$	
1.	$\frac{[F(\text{mar})]}{(F(\text{mar}) \rightarrow D(\text{mar}))}$	
2.	$\frac{D(m) \quad n\text{-}D(m)}{n\text{-}F(m)}$	$R(x) \rightarrow F(x)$
3.	$\frac{n\text{-}F(m)}{n\text{-}R(\text{mar}) \rightarrow n\text{-}R(\text{dat})}$	4. $\frac{R(\text{mar}) \rightarrow F(\text{mar})}{n\text{-}R(\text{mar})}$
	5. $\frac{n\text{-}R(\text{mar}) \rightarrow n\text{-}R(\text{dat}) \quad n\text{-}R(\text{mar})}{n\text{-}R(\text{dat})}$	
6.	$\frac{n\text{-}R(\text{dat})}{n\text{-}R(\text{dat})}$	

Ripercorriamo l'inferenza.

(a) Dalla premessa  $V(x) \quad (F(x) \rightarrow D(x))$  si deduce che  $(F(\text{mar}) \rightarrow D(\text{mar}))$  (se il marito commette un fatto illecito egli commette un danno azionabile), applicando la regola dell'eliminazione del quantificatore universale (VE).

(b) Assumendo che  $F(\text{mar})$  (il marito ha commesso un fatto illecito) deduciamo che  $D(\text{mar})$  (il marito ha commesso un danno azionabile), per la regola dell'eliminazione dell'implicazione (o *modus ponens*).

(c) Questa conclusione è in contraddizione con la premessa  $n\text{-}D(\text{mar})$  (il marito non ha commesso un danno azionabile), e quindi per la regola della *reductio ad absurdum*, possiamo rigettare la premessa  $F(\text{mar})$  e concludere che  $n\text{-}F(\text{mar})$  (il marito non ha commesso un fatto illecito). Si noti che in generale dalle premesse  $n\text{-}B$  e  $A \rightarrow B$  possiamo dedurre  $n\text{-}A$  (*modus tollens*). Questa regola di inferenza non era compresa tra quelle da noi introdotte e quindi abbiamo dovuto costruirla nella nostra argomentazione.

(d) La premessa generale  $R(x) \rightarrow F(x)$  (una persona è responsabile solo se commette un fatto illecito) ci consente di inferire  $R(\text{mar}) \rightarrow F(\text{mar})$  (il marito è responsabile solo se commette un fatto illecito).

(e) Per la regola del *modus tollens* da  $R(x) \rightarrow F(x)$  e  $\neg F(x)$  deduciamo  $\neg R(x)$ .

(f) Infine, la regola del *modus ponens*, applicata a  $\neg R(x)$  e alla premessa n. 4  $\neg R(\text{mar}) \rightarrow \neg R(\text{dat})$  ci consente di dedurre la conclusione  $\neg R(\text{dat})$  (il datore di lavoro non è responsabile).

Si tratta di una deduzione logicamente valida. Ciò non significa che le premesse (e quindi anche la conclusione) debbano essere accettate: le premesse 1. e 3. non incontrano (almeno nel nostro paese) il consenso concorde della dottrina. Non sembra indiscutibile neppure la ricostruzione operata da Tammelo della premessa implicita: proprio in relazione ai casi di responsabilità per fatto altrui si discute se il fatto di cui si tratta debba necessariamente essere un fatto illecito completo di tutti gli elementi che condizionano la responsabilità dell'autore del fatto.

## 2. La programmazione logica

Nelle pagine seguenti si considererà l'uso della programmazione logica (intesa in senso ampio) per la rappresentazione della conoscenza giuridica. Si considereranno tre approcci diversi:

- la programmazione logica in senso stretto, usata presso l'Imperial College:

- la Teoria generale del discorso, un formalismo che comprende l'intera logica classica dei predicati, adottato nel progetto LEX:

- il Linguaggio per il discorso giuridico, un'estensione delle clausole di Horn nella logica intuizionista, sviluppata da McCarty.

### 2.1. La programmazione logica in senso stretto

La programmazione logica in senso stretto, l'uso di formalismi che si richiamano alle clausole di Horn, e in particolare del PROLOG, ha ottenuto un indubbio successo nell'informatica giuridica. L'uso della programmazione logica per rappresentare contesti giuridici è stato approfondito presso l'Imperial College di Londra<sup>27</sup>, dove

---

<sup>27</sup>Cfr. cap. 4, par. 3.1.

sono state avviate le prime esperienze e si sono ottenuti i risultati più significativi. L'esperienza che ha segnato l'inizio del rapporto tra diritto e programmazione logica, è stata la formalizzazione del *British Nationality Act* (normativa in materia di acquisto della cittadinanza inglese)<sup>28</sup>. Su questa esperienza si è aperto un approfondito dibattito sulle prospettive e i limiti dell'uso della programmazione logica, e, in generale, di metodi logici per la rappresentazione e l'uso della conoscenza giuridica. Nelle pagine seguenti si illustrerà la formalizzazione del *British Nationality Act*. Il dibattito sulle prospettive della programmazione logica nel diritto sarà considerato nel par. 6.

### 2.1.1 La formalizzazione del *British Nationality Act*

Per la formalizzazione del *British Nationality Act* fu usato il guscio APES. Si tratta di un meta-interprete che usa la medesima strategia deduttiva del PROLOG, ma facilita la creazione di sistemi basati sulla conoscenza in quanto:

(a) comprende un'interfaccia che consente la scelta tra diverse sintassi del PROLOG e agevola l'immissione delle clausole;

(b) genera spiegazioni del processo inferenziale (spiegazioni che possono essere esaminate anche prima della conclusione finale),

(c) comprende strumenti per l'interrogazione dell'utente (*query the user*), che consentono di definire il formato delle domande senza modificare la rappresentazione logica della conoscenza, di controllare e memorizzare le risposte dell'utente<sup>29</sup>.

Sia in questo lavoro, sia nei successivi contributi del gruppo dell'Imperial College, viene perseguito con coerenza un uso dichiarativo del PROLOG: il PROLOG è usato per la rappresentazione logica<sup>30</sup> della conoscenza giuridica, non per prescrivere particolari processi di elaborazione<sup>31</sup>. Il compito dell'uso della conoscenza resta affidato al motore inferenziale del PROLOG.

---

<sup>28</sup>SERGOT M.J., CORY H.T., HAMMOND P., KOWALSKI R.A., KRIWACZEK F., SADRI F., *Formalisation of the British Nationality Act*, cit.

<sup>29</sup>Cfr. cap. 7 par. 2.1.

<sup>30</sup>Nel sottoinsieme della logica dei predicati costituito dalle clausole di Horn estese con la negazione come fallimento.

<sup>31</sup>Ci siamo già soffermati (cap. 4, par. 3.1) sulla differenza tra uno procedurale e dichiarativo del PROLOG.

Ad esempio, consideriamo la formalizzazione del primo articolo della legge<sup>32</sup>:

a) Testo originale:

1.(1) Una persona nata nel Regno Unito dopo l'entrata in vigore è un cittadino inglese se al momento della nascita il padre o la madre è

- (a) cittadino inglese, o
- (b) residente nel regno unito.

1.(2) Un neonato che, dopo l'entrata in vigore, è stato trovato abbandonato nel Regno Unito, (a meno che non sia provato il contrario) è considerato ai fini della sottosezione (1)

- (a) nato nel Regno unito dopo l'entrata in vigore e
- (b) nato da un genitore che al momento della nascita era un cittadino inglese o risiedeva nel Regno Unito.

b) Rappresentazione in clausole di Horn.

Il primo articolo del British Nationality Act veniva formalizzato come segue in clausole di Horn. Per lo sviluppo di una base di conoscenza, la disposizione doveva essere riformulata in più clausole PROLOG<sup>33</sup>.

x acquista la cittadinanza inglese alla data y in base alla sezione 1.1 se

- x è nato nel Regno unito e
- x è nato nella data y e
- y è successivo o contemporaneo all'entrata in vigore e
- x ha un genitore qualificato a norma della sezione 1.1 alla data y.

<sup>32</sup>SERGOT M.J., CORY H.T., HAMMOND P., KOWALSKI R.A., KRIWACZEK F., SADRI F., *Formalisation of the British Nationality Act*, cit., p. 372.

<sup>33</sup>Le ragioni di una tale ristrutturazione sono ampiamente discusse nel lavoro citato nella nota precedente.

x acquista la cittadinanza inglese alla data y in base alla sezione 1.2 se  
x è stato trovato come neonato abbandonato nel Regno unito e  
x è stato trovato nella data y e  
y è successivo o contemporaneo all'entrata in vigore e  
not [x non è stato trovato nel Regno unito dopo o durante l'entrata in vigore] e  
not [x non è nato da un genitore qualificato a norma della sezione 1.1 al momento della nascita].

x ha un genitore qualificato a norma della sezione 1.1 alla data y se  
Z è genitore di x e  
x è un cittadino inglese alla data y

x ha un genitore qualificato a norma della sezione 1.1 alla data y se  
Z è genitore di x e  
x risiede nel Regno Unito nella data y

Si osservi che not esprime una negazione come fallimento (significa: "non è dimostrabile che"), mentre non esprime la negazione classica (significa "non è vero che"). Inoltre, vengono indicare espressamente certi aspetti temporali che restano impliciti nel testo di legge (il fatto che la data della nascita sia anche quella sia anche quella dell'acquisto della cittadinanza ecc.). Il concetto di "avere un genitore qualificato a norma della sezione 1.1" viene definito a parte, in modo da ottenere una rappresentazione più compatta.

Viene introdotta una clausola ulteriore per esprimere un assunto implicito<sup>34</sup>: la dopo la morte o la perdita della cittadinanza una persona non può più essere qualificato come cittadino.

x è un cittadino inglese alla data y in base alla sezione z se  
x è vivo alla data y e  
x acquista la cittadinanza inglese alla data y<sup>1</sup> in base alla sezione Z e  
y è successiva o contemporanea a y<sup>1</sup> e  
not [x cessa di essere un cittadino inglese nella data y2 e y2 è tra y1 e y].

<sup>34</sup>Questo principio corrisponde a quello illustrato nel par. 1.2.1 a).



## 2.2. La teoria generale del discorso

Nel sistema LEX la conoscenza giuridica è stata rappresentata in un formalismo ispirato alla logica predicativa, la teoria generale del discorso (*General Discourse Theory-GDT*). La teoria della rappresentazione del discorso è un'estensione della logica predicativa (nel senso che consente di rappresentare anche eventi, stati e contesti temporali) particolarmente adatta a fungere da punto d'arrivo per l'analisi automatica del linguaggio naturale<sup>35</sup>. Infatti, con questo formalismo si è inteso, rispondere alla duplice esigenza di rappresentare la conoscenza in un linguaggio logico e di trattare il linguaggio naturale. Nel progetto LEX si è definita una rappresentazione delle modalità deontiche nella GDT<sup>36</sup>.

"Le nozioni fondamentali di questa teoria sono:

1. Strutture della rappresentazione del discorso (*Discourse Representation Structures - DRSs*): Una DRS è una coppia  $\langle U, C \rangle$ , dove  $U$  è un insieme di contrassegni di riferimento (*reference markers*) e  $C$  è un insieme di condizioni, e

2. Algoritmi per la costruzione di DRSs che traducono il linguaggio naturale in DRSs.

Le condizioni sono atomiche, cioè della forma  $P(u^1, \dots, u^n)$  o complesse. Le condizioni complesse sono:

1. condizioni implicative:  $K^1 \rightarrow K^2$
2. condizioni disgiuntive:  $K^1 \vee K^2$
3. condizioni negative:  $\neg K^1$
4. condizioni di evento:  $e:K$ .

Il vantaggio fondamentale della teoria generale del discorso (GTD) consisterebbe "nel suo preciso fondamento logico e pertanto nella sua semantica ben definita e nella teoria deduttiva corrispondente. La GTD comprende l'intera logica dei predicati"<sup>37</sup>, e si presta al trattamento degli aspetti sintattici e semantici di testi in linguaggio

<sup>35</sup>KAMP H., *A Theory of Truth and Semantic Representation*, in GROENENDIJK J.A.G., JANSSEN T.M.V., STOKHOF M.B.G., *Formal Methods in the Study of Language*, MC TRACT 135, Eds. Amsterdam, 1981, pp. 277-322; KAMP H., *Discourse Representation Theory: What it Is and where it ought to Go*, in BLAZER A. (a cura di) *Natural Language and the Computer*, cit. pp. 84-111.

<sup>36</sup>LEHMANN H., ZOEPPRITZ M., *Formale Behandlung der Begriffs 'Pflicht'*, cit.

<sup>37</sup>Cfr. LEHMANN H., *The LEX-Project - Concepts and Results*, cit., p. 121. La sintassi delle strutture della rappresentazione del discorso si avvicina a quella della logica predicativa.

naturale, e, particolare, al trattamento dei riferimenti contestuali.

In un primo prototipo (LEX0) la conoscenza era stata rappresentata direttamente in PROLOG. Circa le ragioni che hanno indotto a passare alla GDT, Lehmann osserva che il le clausole di Horn presentano il vantaggio di una relativa efficienza computazionale, al prezzo però di un difetto di capacità espressiva. Secondo Lehmann questo difetto può essere superato con un trattamento procedurale della conoscenza, ma l'uso di un trattamento procedurale della conoscenza comporterebbe non solo una perdita di chiarezza e di rigore logico, ma renderebbe più difficile il passaggio dal linguaggio naturale al linguaggio formale<sup>38</sup>.

Per illustrare i vantaggi della GDT, Lehmann prende in considerazione il seguente enunciato:

**"Un incidente stradale è un evento improvviso, causato dai pericoli del traffico stradale e che causa danni notevoli".**

In LEX0 (il prototipo in PROLOG), la formalizzazione era stata la seguente:

```
incidente(I,A) se
    evento_improvviso(U) e
    causa_di_pericoli_d_traff(U) e
    causò_danni_notevoli(I,A),!.
```

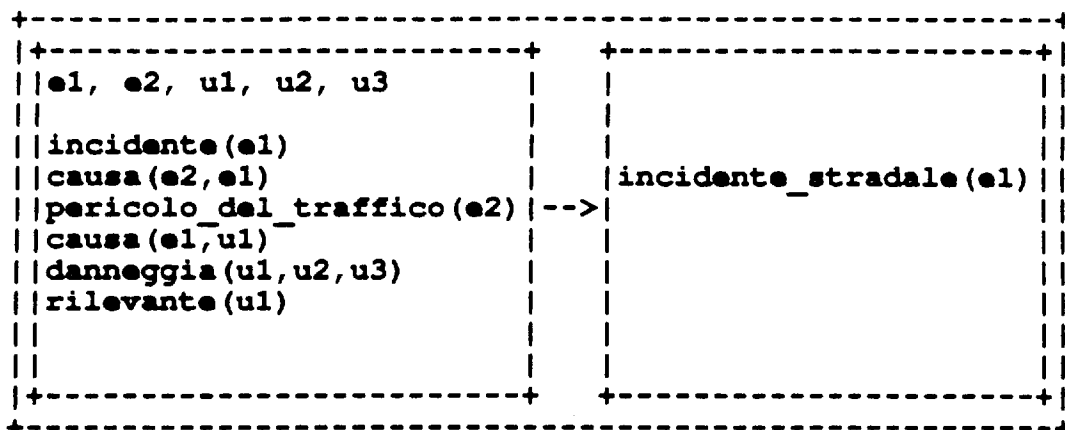
che significa che vi è un incidente I che causa un danno per l'ammontare A, se sono soddisfatte le 3 condizioni indicate dai predicati **evento improvviso**, **causa di notevoli pericoli d traff**, **causò danni notevoli**, predicatī che sono definitī in ulteriori clausole di Horn<sup>39</sup>.

Il medesimo enunciato è stato espresso come segue nella teoria generale del discorso:

---

<sup>38</sup>LEHMANN H., *Formale Rāpresentation juristisches Wissens*, cit., p. 78-79. Dobbiamo osservare, a questo riguardo, che sinora, nessun motore inferenziale in grado di trattare l'intera logica dei predicati ha dato risultati soddisfacenti. Anche i risultati ottenuti nell'ambito di LEX non sembrano decisivi.

<sup>39</sup>Cfr. LEHMANN H., *The LEX-Project - Concepts and Results*, cit., p. 125 s.



Questa formulazione indica che

```

se l'evento e1 è un incidente e
    ha come causa l'evento e2 e
    e2 è un pericolo del traffico e
    e1 causa u1 e
    u1 è un danno e
    u1 è rilevante
allora
e1 è un incidente stradale.

```

Lehmann osserva che "in confronto all'esposizione in PROLOG i predicati usati nella nelle DRSS sono da un lato più vicini ai concetti del linguaggio naturale, d'altro lato più generali. Ciò può rendere più pesante l'elaborazione nel sistema informatico, ma conduce a risultati intuitivamente soddisfacenti"<sup>40</sup>.

La base di conoscenza di LEX comprende diversi tipi di conoscenze. Le conoscenze di livello più elementare sono rappresentate con formalismi diversi dalle DRTs. Si tratta di:

- una base di dati che comprende circa 16.000 voci del linguaggio comune e del linguaggio giuridico e riporta le caratteristiche morfologiche e grammaticali di norme, verbi e aggettivi;
- un *thesaurus* che rappresenta una gerarchia concettuale, ad esempio:

```

bt('accusato','essere.umano')
bt('essere.umano','essere.vivente')
bt('essere.vivente','oggetto.naturale')41

```

<sup>40</sup>LEHMANN H., *Formale R presentation juristisches Wissens*, cit., p. 80.

<sup>41</sup>bt significa termine pi  generale (*broader term*).

-una rappresentazione delle restrizioni concettuali di alcuni verbi. Ad esempio:

```
verb(subire,1, evento,
      nom(dist('essere.umano')
      acc(dist,'danno'.'perdita','incidente'))
```

indica che il verbo "subire" ha una sola lettura (non è ambiguo), rappresenta un evento (nel sistema si distingue tra verbi di stato e di evento) e specifica il soggetto (nominativo) è un essere umano mentre il complemento oggetto (accusativo) è un danno, una perdita o un incidente.

Le conoscenze più complesse sono rappresentate invece come DRTs. La base di conoscenza contiene conoscenze del senso comune e conoscenze giuridiche.

### 2.2.1 Conoscenze del senso comune

Concetto di collisione:

<pre>u<sup>1</sup>,u<sup>2</sup>,e<sup>1</sup>: colpire(u<sup>1</sup>,u<sup>2</sup>,u<sup>3</sup>,e<sup>1</sup>) cade(u<sup>2</sup>,e<sup>1</sup>)</pre>	---->	<pre>investire2(u<sup>2</sup>,u<sup>1</sup>,e<sup>1</sup>)</pre>
--	-------	--

### 2.2.2 Conoscenze giuridiche.

Un interessante aspetto di LEX è tentativo di collegare conoscenze del senso comune e conoscenze giuridiche, così da render conto della parziale tecnicizzazione del linguaggio giuridico. Per distinguere vocaboli del linguaggio comune e linguaggi giuridici, aggiungiamo il suffisso j:

<pre>e<sup>1</sup>,u<sup>1</sup>: improvviso_j(e<sup>1</sup>,u<sup>1</sup>)</pre>	improvviso(e <sup>1</sup> )
---	-----------------------------

(se un evento è improvviso in senso giuridico, allora è un evento improvviso).

$u^1, u^2, u^3, e^1:$ $\text{danno\_a\_persona}(u^1, u^2, u^3, e^1)$ $\text{danno\_a\_cosa}(u^1, u^2, u^3, e^1)$	$\rightarrow$	$\text{danno\_j}(u^1, u^2, u^3, e^1)$
--	---------------	---------------------------------------

(se un evento è un danno ad una persona o è un danno ad una cosa, allora è un danno in senso giuridico).

Si è usata la teoria generale del discorso anche per rappresentare concetti deontici<sup>42</sup>. La relazione tra un Obbligo  $o$ , un soggetto passivo  $p$  e un soggetto attivo  $a$  viene rappresentata con un predicato a tre argomenti  $\text{Obbligo}(o, p, a)$ . Vediamo come la seguente situazione giuridica può essere rappresentata in DRSs:

**Una persona coinvolta nell'incidente ha verso le altre persone danneggiate o coinvolte nell'incidente il dovere di consentire ispezioni**

L'azione prescritta è indicata introducendo un ulteriore predicato  $\text{soddisfa}(p, o)$  (il soggetto passivo  $p$  soddisfa l'obbligo  $o$ ), definito come segue:

$e, o, a$ $e: \text{soddisfa}(p, o)$	$\leftrightarrow$	$K$
---	-------------------	-----

dove  $K$  è una struttura della rappresentazione del discorso che descrive l'azione prescritta. La struttura sopra rappresentata indica che si verifica un evento  $e$  consistente nella soddisfazione dell'obbligo  $p$  da parte di  $a$  se e solo se si verifica  $K$ . Ad esempio

<sup>42</sup>LEHMANN H., *Formale R presentation juristisches Wissens*, cit., p. 80.



verificava per i linguaggi della rappresentazione della conoscenza della fine degli anni 70"<sup>45</sup>.

McCarty sta sviluppando una implementazione (un metainterprete) del linguaggio per il discorso giuridico in Golden Common LISP<sup>46</sup>, su un minielaboratore (stazione di lavoro SUN/3)<sup>47</sup>. Presso il CIRFID dell'Università di Bologna si intende sviluppare invece un metainterprete per LLD in PROLOG, su minielaboratore.

Nelle pagine seguenti esamineremo brevemente i contenuti fondamentali della proposta di McCarty, che rappresenta il tentativo più articolato di realizzare un linguaggio per la rappresentazione della conoscenza che tenga conto sia della complessità della conoscenza giuridica sia dei problemi dell'efficienza computazionale.

LLD si basa sulla logica intuizionistica. La principale giustificazione della proposta di questo linguaggio è infatti la seguente: adottando la logica intuizionistica sarebbe possibile ottenere un'estensione del linguaggio delle clausole di Horn che possieda una capacità espressiva sufficiente per rappresentare la conoscenza giuridica, ma conservi i vantaggi computazionali delle procedure di inferenza per le clausole di Horn. Se la medesima estensione fosse interpretata nella logica classica quei vantaggi andrebbero inevitabilmente perduti. Non ci si dilungherà oltre sulla difficile problematica del rapporto tra clausole di Horn, logica classica e logica intuizionistica<sup>48</sup>, e si illustreranno brevemente le principali caratteristiche di LLD.

### 2.3.1 La sintassi delle formule atomiche

Le formule di LLD hanno una sintassi interna, usata dalle procedure di inferenza (dal motore inferenziale), e una sintassi superficiale, per la comunicazione con l'utente esterno (l'ingegnere della conoscenza). Si considerino i seguenti esempi di sintassi interna:

---

<sup>45</sup>MCCARTY T., *A Language for Legal Discourse. I Basic Features*, cit.

<sup>46</sup>Una delle versioni più diffuse del linguaggio LISP (cf. cap. 1, par. 3.2).

<sup>47</sup>Cfr. cap. 4 par. 2.1.

<sup>48</sup>Su questa problematica, si rinvia a MCCARTY T., *Clausal Intuitionistic Logic. I. Fixed Point Semantics*, in "Journal of Logic Programming", 5 (1), 1988, pp. 1-31; MCCARTY T., *Clausal Intuitionistic Logic. II. Tableau Proof Procedures*, in "Journal of Logic Programming", 5 (2), 1988, pp. 93-132;

**(Possiede P1 (Persona A) (Bene B)**

(vi è una relazione di possesso P1 tra una persona A e un bene B).

**(Possiede 'Possiede-2 (Persona 'Giovanni) (Bene 'Libro-3)**

(la relazione Possiede-2 sussiste tra la persona Giovanni e il bene Libro-3).

**(Possiede - (Persona 'Giovanni) (Bene 'Libro-3)**

(vi è una relazione di possesso tra la persona Giovanni e il bene Libro\_3<sup>49</sup>)

Nella sintassi superficiale vengono aggiunti degli identificatori agli argomenti:

**(Possiede 'Possiede-2 {soggetto (Persona 'Giovanni) oggetto (Bene 'Libro-3).**

Osserviamo che in LLD ogni relazione è trattata come un oggetto individuale, anche le relazioni sono considerate individui dell'universo del discorso, e vengono rappresentate con variabili o costanti (come **possiede 2**). Questa reificazione delle relazioni è ampiamente usata nel linguaggio comune, e anche nel linguaggio giuridico (si pensi ad affermazioni come le seguenti: il rapporto obbligatorio tra P1 e P2 è stato costituito con l'atto A, è stato risolto con l'atto B, si è trasformato nel rapporto C ecc.).

Tutti i termini di LLD appartengono ad un determinato tipo, e i tipi sono ordinati in una gerarchia (ad esempio, una **Persona** può essere una **Persona fisica** o una **Persona giuridica**)<sup>50</sup>. Le procedure di inferenza, e, in particolare, l'algoritmo di unificazione rispettano i tipi degli oggetti trattati.

Si distingue inoltre tra termini contabili (come **Persona**, **Cosa** ecc.) e termini di massa (come **Capitale**, **Denaro**, **Acqua** ecc.). I termini di massa sono trattati come un insieme infinito di particelle infinitesimale e ad essi possono essere associate una o più misure

---

<sup>49</sup>La sintassi adottata da McCarty per distinguere tra variabili e costanti differisce da quella usata nel presente volume. In particolare, le costanti sono precedute da un apice "'", e questa caratteristica le distingue dalle variabili. Il trattino - indica la variabile anonima, è cioè un'espressione che sta per una variabile qualsiasi.

<sup>50</sup>Sul tipo della gerarchia è un che abbiamo introdotto nel cap. par.



Denaro P  
 (soggetto di (Valore -  
   {unità (lira - )}  
   {quantità (numero N)}})).

(la somma di denaro P che ha il valore di N lire)

### 2.3.2 La sintassi della formule molecolari

#### a) Le clausole di Horn

Le clausole di Horn costituiscono una struttura fondamentale per la rappresentazione della conoscenza<sup>51</sup> anche in LLD. Nella sintassi di LLD<sup>52</sup> le clausole di Horn si rappresentano come segue:

```
(Controlla -
  {soggetto (Persona P)}
  {oggetto (Società S)})

<== (Possiede -
      {Soggetto (Persona P)}
      {Oggetto (Capitale_azionario C)})

AND

(ha-emesso -
  {Soggetto (Società S)}
  {Oggetto (Capitale_azionario A)})
```

(una persona controlla una società se possiede il capitale azionario emesso da quella società).

#### b) Negazioni e implicazioni annidate

Una prima estensione del linguaggio delle clausole di Horn è rappresentato dalla possibilità di usare clausole di Horn annidate nel lato destro di una regola. Cioè le condizioni di una regola, anziché essere enunciati atomici, possono essere clausole di Horn. Esempio:

---

<sup>51</sup>La logica delle clausole di Horn è la stessa nell'interpretazione della logica classica (adottata ad esempio, da Kowalski e Sergot o nel progetto Lex) e nella logica intuizionista.

<sup>52</sup>MCCARTY T., *A Language for Legal Discourse. I Basic Features*, cit., p. 4.

(Sterile {oggetto (contenitore C)})

<== PER OGNI (Insetto I)):   
 morto - {oggetto (Insetto I)}

<==(Dentro - {soggetto (Insetto I)}   
 {oggetto (Contenitore C)}).

(un contenitore è sterile se, ogni insetto è morto qualora sia contenuto in C).

Le negazioni vengono rappresentate utilizzando questa struttura sintattica. Si tratta della negazione intuizionistica che differisce dalla negazione come fallimento usata nelle clausole di Horn. Esempio:

res-nullius - {oggetto (Bene B)}

<== PER OGNI (Persona P):

FALSO <==

(Possiede - {soggetto (Persona P)}   
 {oggetto (Bene B)}).

(Un bene è una *res nullius* se il fatto che qualcuno lo possiega implica una contraddizione)<sup>53</sup>

### c) Il ragionamento per difetto

Il ragionamento per difetto è trattato con un operatore *fail*, che ha un significato vicino alla negazione come fallimento del PROLOG, ma può essere combinato con la negazione intuizionistica<sup>54</sup>. Ad esempio "per esprimere che <<un adulto si presume essere occupato a meno che non si dimostri che è un emarginato>> useremo la seguente regola

(Occupato - {oggetto (Persona P)})

<== (Adulto - {oggetto (Persona P)})   
 AND FAIL NOT NOT   
 Emarginato - {oggetto (Persona P)})

<sup>53</sup>Cioè, se nessuno lo possiede.

<sup>54</sup>MCCARTY T., *A Language for Legal Discourse. I Basic Features*, cit., p. 5.

#### d) Prototipi e deformazioni

In LLD McCarty riprende la teoria di prototipi e trasformazioni<sup>55</sup>, ma in un quadro concettuale profondamente mutato.

McCarty afferma che, usando il modello di prototipo e trasformazioni, è possibile ottenere una capacità espressiva simile a quella consentita da asserzioni disgiuntive o esistenziali (non previste in LLD). In questo modo sarebbe possibile rappresentare anche definizioni del tipo se e solo se (che stabiliscono condizioni necessarie e sufficienti), senza usare clausole con conclusione disgiuntiva<sup>56</sup>. Infatti McCarty suggerisce di considerare uno dei disgiunti che rappresentano condizioni necessarie e sufficienti come prototipico e di trattare gli altri disgiunti come trasformazioni del prototipo. Consideriamo il seguente esempio:

**(Pacco\_Di\_Natale P)**

**<== (Pacco P) AND**

**[(Colorato -  
  {oggetto (Pacco P)}  
  {qualità (Colore 'Rosso)})]**

**OR**

**[(Colorato -  
  {oggetto (Pacco P)}  
  {qualità (Colore 'Verde')}]**

(B è un pacco di Natale se B è un pacco rosso oppure B è un pacco verde).

Supponiamo però di voler esprimere l'affermazione più forte che (B è un pacco di Natale se e solo se B è un pacco rosso oppure B è un pacco verde)

Il connettivo (se e solo) dovrebbe essere rappresentato aggiungendo una regola con conclusione disgiuntiva alla regola precedente.

---

<sup>55</sup>Cfr, cap. 4 par. 2.1.

<sup>56</sup>L'enunciato  $A \leftrightarrow B \vee C$ , equivale infatti alle due clausole  $A \leftrightarrow B \vee C$  e  $B \vee C \leftrightarrow A$ .

(Pacco P) AND

```
([(Colorato -
  {oggetto (Pacco P)}
  {qualità (Colore 'Rosso)})]
```

OR

```
[(Colorato -
  {oggetto (Pacco P)}
  {qualità (Colore 'Verde)})]
```

<== (Pacco\_Di\_Natale P)

Grazie a questa regola, se un pacco è un pacco di Natale potremmo dedurre che il pacco è rosso o verde. Regole di questo tipo pregiudicherebbero però l'efficienza della dimostrazione automatica<sup>57</sup>. La soluzione suggerita da McCarthy consente di considerare una delle due conseguenze della seconda regola (ad esempio l'ipotesi che il pacco sia rosso) come prototipica. Si potrebbe passare ad esaminare l'altra conseguenza (l'ipotesi che il pacco sia verde) solo se l'ipotesi prototipica conduca ad una contraddizione. In questo caso si applicherebbe la trasformazione 'rosso-verde' alla prova prototipica.

### 2.3.3 Le modalità

Il linguaggio proposto da McCarthy include una rappresentazione di concetti modali.

#### a) Il tempo

I concetti temporali sono espressi segue:

```
(Stato -
  {relazione
    (Controllo
      {soggetto (Società A1)}
      {oggetto (Società A2)}})
  {tempo (Tempo T)}).
```

(Nel tempo T sussiste una relazione di controllo tra le società A1 e A2)

---

<sup>57</sup>Poiché, quando si sia verificata la premessa il sistema dovrebbe sviluppare non più linee di ragionamento diverse, senza sapere quale sia la corretta.

```

(Stato -
  {relazione
    (Controllo
      {soggetto (Società A1)}
      {oggetto (Società A2)}})
  {tempo (Tempo T1)}
  {tempo (Tempo T2)}}).

```

(Nell'intervallo tra T1 e T2 sussiste una relazione di controllo tra le società A1 e A2)

#### b) Eventi ed Azioni

Gli eventi elementari sono rappresentati come cambi di stato, cioè come il passaggio da uno stato ad un altro stato incompatibile con il primo (eventi complessi possono essere costruiti combinando eventi elementari), mentre il concetto di azione è formulato nei termini della relazione tra soggetti ed eventi,

```

(Cambio di stato -
  {relazione1
    (Possiede O1
      {soggetto (Persona P1)}
      {oggetto (Bene B)}})
  {relazione2
    (possiede O2
      {soggetto (Persona P2)}
      {oggetto (Bene B)}})
  {tempo (Tempo T1)}
  {tempo (Tempo T2)}}).

```

(Si è verificato l'evento consistente nel passaggio dal possesso del bene B da parte di P1 nel tempo T1 al possesso di B da parte di P2 nel tempo T2 (un trasferimento di proprietà)).

#### c) I concetti deontici

Senza approfondire la logica deontica proposta da McCarty<sup>58</sup>, si riporta un enunciato deontico nella sintassi di LLD:

---

<sup>58</sup>Per la quale rinviamo a MCCARTY T., *Permission and Obligation: an Informal Intruduction*, cit.

```

(Obbligazione
  {condizione
    {emettere -
      {soggetto (Società S)}
      {oggetto (Azione_Ordinaria A)}}}
  {azione
    (DistribuireDividendi -
      {agente (società S)}
      {oggetto (Denaro &D)}}}
    {soggetto di (Valore -
      {unità (lira - )}
      {quantità (numero N)}})})
  {ricevente (Persona &P
    {soggetto di
      {possedere -
        {oggetto
          (Azione_Ordinaria A)}}})}
    {tempo (Tempo T1)}
    {tempo (Tempo T2)}}
  tempo (Tempo T3))).

```

(Nel tempo T3 sussiste l'obbligo che qualora la società S emetta nel tempo T1 l'azione ordinaria A, la società S distribuisca nel tempo T2 il denaro D alla persona P che possiede l'azione ordinaria A.)

#### d) Conclusione

Nelle pagine precedenti ci si è limitati ad illustrare alcuni aspetti della sintassi del linguaggio per il discorso giuridico di McCarty. Per poter valutare il senso di questa proposta occorrerebbe esaminare la semantica e le procedure di inferenza associate a questa sintassi. Ma ciò richiederebbe un approfondimento della logica intuizionistica e della teoria della programmazione logica, problematiche che non è possibile affrontare in questa sede. Possiamo tuttavia affermare che LLD rappresenta la proposta più organica e completa per la rappresentazione della conoscenza giuridica. Se sarà possibile una efficiente implementazione, alcuni importanti problemi della rappresentazione formale della conoscenza giuridica potranno trovare soluzione<sup>59</sup>. Non mancano i punti oscuri. In particolare il concetto di prototipo e deformazione non sembra abbia ancora raggiunto un grado di elaborazione sufficiente a consentire un'implementazione informatica.

---

<sup>59</sup>La proposta di McCarty rappresenta, sotto vari aspetti, lo sviluppo di alcune critiche rivolte all'uso delle clausole di Horn (la programmazione logica in senso stretto). Cfr. par. 6.

### 3. Le regole di produzione

A prima vista, le regole di produzione sembrano particolarmente adatte alla rappresentazione di contesti giuridici. L'affermazione che il diritto si compone di norme o regole sembra un'ovvietà. Tuttavia, prima di esaminare in quale misura il diritto sia rappresentabile in regole di produzione, bisogna precisare il rapporto tra il concetto giuridico di regola o norma (e, in particolare, di norma condizionale o ipotetica) come enunciato prescrittivo e il concetto informatico di regola di produzione.

#### 3.1. Le norme ipotetiche

Numerosi filosofi e teorici del diritto hanno affermato che le norme giuridiche sono rappresentabili come enunciati condizionali (o ipotetici). Per alcuni tutte le norme giuridiche rappresenterebbero, o comunque sarebbero riducibili a, un tipo determinato di enunciati condizionali, cioè ad enunciati che esprimono *norme ipotetiche*.

Secondo la nota concezione di Kelsen le norme giuridiche avrebbero una struttura del tipo:

**se <illecito> allora deve essere <sanzione>,**

dove la locuzione "deve essere" esprime il dover essere, il particolare rapporto tra illecito e sanzione, il cosiddetto rapporto di imputazione, che Kelsen contrappone alla causalità<sup>60</sup>; ad esempio,

**"Se un individuo commette omicidio, il tribunale a ciò autorizzato deve comminare contro di lui la pena di morte"**<sup>61</sup>.

---

<sup>60</sup>Nelle ultime opere di Kelsen, il concetto di imputazione viene esteso a designare non solamente il rapporto tra illecito e sanzione, ma qualsiasi rapporto tra fattispecie e conseguenza giuridica. Cfr. KELSEN, *La dottrina pura del diritto*, cit. "il dovere giuridico il quale è il verbo modale che, nella proposizione giuridica, collega la prescrizione con l'oggetto, abbraccia tutti e tre i significati: quello di prescrizione, quello di autorizzazione e quello di permissione positiva della conseguenza". L'affermazione di Kelsen si riferisce alle proposizioni giuridiche, cioè agli enunciati che descrivono delle norme (e non alle norme stesse), ma può essere trasposta anche alle norme, tenuto conto del rapporto di "isomorfismo" tra norme e proposizioni giuridiche affermato da Kelsen (almeno in questa fase del suo pensiero).

<sup>61</sup>Cfr. KELSEN, *La dottrina pura del diritto*, cit., p. 71.

Accanto a questo concetto di norma, dove l'operatore deontico sembra assorbito dal connettivo "allora deve essere", nell'opera di Kelsen<sup>62</sup>, emerge una rappresentazione della norma ipotetica diversa e più accettabile:

**se <condizione> allora <operatore deontico>  
(<fatto\_qualificato>).**

Si osservi che la distinzione che si è proposta prescinde dal fatto che per Kelsen le norme siano rivolte ai giudici. Si tratta della questione se nelle norme condizionali (a chiunque rivolte) l'operatore deontico si includa nel connettivo (nel dover essere dell'imputazione) o sia una qualificazione dell'enunciato conseguente. A mio parere, è preferibile la seconda soluzione, poiché (a) è utile poter rappresentare non solo norme che impongono degli obblighi, ma anche norme permissive e attributive di poteri e (b) non sembra ragionevole definire un diverso tipo di implicazione per ciascuna qualificazione deontica.

Una versione più aggiornata del concetto di norma giuridica come norma ipotetica è stato proposto da Alchourron e Bulgin, secondo i quali ogni norma giuridica collega un caso ad una soluzione, ed ha quindi una struttura del tipo

**se <caso> allora <soluzione>,**

dove <caso> è una combinazione di proprietà o circostanze rilevanti e <soluzione> è la qualificazione deontica di una combinazione di azioni come ad esempio:

**se <il possessore attuale (l'avente causa) è in mala fede> allora <egli ha l'obbligo di restituire la cosa al proprietario><sup>63</sup>.**

Un'altra versione di questo concetto di norma, specificamente riferito a problematiche informatico-giuridiche, è stata proposta da Allen, secondo il quale le norme sono riducibili alla struttura

---

<sup>62</sup>Cfr. KELSEN, *Teoria generale delle norme*, cit., p. 95. Il pensiero di KELSEN a questo riguardo è assai oscuro.

<sup>63</sup>Cfr. ALCHOURRON C., BOULYGIN E., *Normative system*, cit., pag. 7-43 e 181-191, l'esempio è riportato a pag. 15; Cfr. anche ALCHOURRON C., *Conditionality and the Representation of Legal Norms*, in MARTINO A.A., SOCCI NATALI F. (a cura di) *Automated Analysis of Legal Texts*, cit., pp. 175-186.



**se <insieme antecedente di condizioni> allora  
<conseguente deontico>**

ad esempio

**SE una persona è un passeggero  
ALLORA quella persona deve uscire da questo cancello<sup>64</sup>.**

Sono state proposte rappresentazioni delle norme ipotetiche non immediatamente riconducibili al modello della regola di produzione. Così, ad esempio, Von Wright ha recentemente proposto di adottare non lo schema.

**"se <condizione> allora è obbligatorio  
che<conseguenza>",**

in formula:

**<condizione> -> O<conseguenza>,**

ma lo schema:

**"è obbligatorio che, se condizione, allora conseguenza",**

in formula

**O(<condizione> -> <conseguenza>),**

come ad esempio:

**"è obbligatorio che, se fai una promessa, tu la  
rispetti"<sup>65</sup>.**

Una rappresentazione analoga delle norme condizionali è stata recentemente proposta da McCarty, il quale, seguendo la tradizione delle logiche deontiche diadiche (nelle quali l'operatore deontico ha per argomento non una proposizione qualsiasi, ma una coppia del tipo <condizione><conseguenza>), propone una rappresentazione del tipo:

---

<sup>64</sup>ALLEN L.A., SAXON C.S., *Analysis of the Logical Structure of Legal Rules by a Modernized and Formalized Version of Hohfeld's Fundamental Legal Conceptions*, cit., p. 418.

<sup>65</sup>WRIGHT VON, *Bedingungsnormen. Ein Prüfstein für die Logik der Normen*, in KRAVIETZ W., SCHELSKI H., WINKLER G., SCHRAMM A. (a cura di), *Theorie der Normen. Festgabe für Ota Weinberger zum 65 Geburtstag*, Duncker & Humblot, Berlin, 1984, pp. 447-456.

$O(\langle \text{condizione} \rangle / \langle \text{azione} \rangle)$

dove l' $\langle \text{azione} \rangle$  è obbligatoria nella  $\langle \text{condizione} \rangle$ , come ad esempio

(è obbligatorio che  
 ((possiede(società x, il denaro y) /  
 distribuisce\_dividendo(società x))),

cioè

è obbligatorio che qualora la società x possieda denaro, distribuisca un dividendo<sup>66</sup>.

A mio avviso, la rappresentazione della norma ipotetica che stabilisce un'obbligo (o un permesso) condizionato, come una proposizione condizionale che ha per antecedente una certa fattispecie e per conseguente la qualificazione deontica di un fatto è convincente ed intuitiva<sup>67</sup>. Tuttavia, la peculiarità dell'implicazione normativa può consigliare uno specifico trattamento, e a questo fine possono risultare utili formalismi come quello appena presentato.

Anche assumendo di accettare una rappresentazione del tipo  $A \rightarrow O(B)$  per le norme ipotetiche, si pongono però due ulteriori questioni, al fine di determinare i limiti nei quali sia possibile o utile rappresentare le norme giuridiche come regole di produzione:

(a) In quale misura tutto il diritto è riducibile a norme che stabiliscono qualificazioni deontiche condizionate?;

(b) In quale misura il diritto è riconducibile a proposizioni ipotetiche?.

### 3.2. Il diritto è riducibile ad un insieme di obblighi condizionati?

Se tutto il diritto consiste di norme condizionali e tali norme sono proposizioni ipotetiche direttamente esprimibili

---

<sup>66</sup>MCCARTY, *Permission and Obligation* (1986), l'esempio è ripreso da pag. 323.

<sup>67</sup>La principale obiezione contro questa rappresentazione è di tipo logico-filosofico. I formalismi di questo tipo collegano con un connettivo logico una proposizione aleutica (cui è possibile ascrivere valori di verità) e una proposizione normativa, che esprime una qualificazione deontica (cui tali valori non si possono ascrivere, secondo molti studiosi). Pertanto non sarebbe possibile l'interpretazione vero-funzionale dei connettivi logici.

mediante regole di produzione, allora sembra che il problema della scelta delle strutture formali nelle quali rappresentare contenuti giuridici possa avere una semplice soluzione, l'uso di regole di produzione per esprimere norme ipotetiche.

Nella teoria del diritto, è controverso se tutte le norme siano riducibili a regole con un antecedente che descrive una certa situazione e un conseguente che esprime la qualificazione di un fatto secondo una modalità deontica. In particolare, si è affermato che sfuggirebbero a questa rappresentazione le c.d. norme costitutive, oggetto di tante discussioni negli ultimi anni<sup>68</sup>.

In questo contesto non è necessario approfondire il problema. Basta osservare che una rappresentazione di tutto il diritto con regole di produzione che esprimano norme ipotetiche, anche se fosse possibile in astratto<sup>69</sup>, è estremamente inefficiente: vengono meno quei concetti che fungono da termine medio tra le descrizioni dei fatti e le conseguenze deontiche, con la conseguenza di un'incontrollabile esplosione casistica<sup>70</sup>.

Basta sfogliare un qualsiasi testo legislativo o dottrinale per accorgersi che sono una minoranza le norme riducibili al semplice schema

**se <condizione> allora <conseguenza deontica>.**

Infatti, spesso le norme giuridiche connettono ad una determinata condizione uno stato o una qualificazione di cose, persone, atti, dalla quale dipende un insieme di conseguenze giuridiche, altre volte la determinazione del comportamento obbligatorio passa attraverso (l'ordinamento) di istituzioni di vario tipo (il problema delle persone giuridiche), o l'esercizio di poteri normativi delegati.

I concetti giuridici sono collegati in molteplici rapporti, alcuni dei quali sono gerarchici. Ad esempio, le gerarchie del tipo è un sono usate frequentemente per estendere alla specie la disciplina del genere, salva la possibilità di diversa disciplina specifica (ad esempio, ai contratti nominati, come la compravendita, è applicabile la disciplina generale dei contratti, in quanto non sia

---

<sup>68</sup>Cfr. CONTE A., *Deontico vs. dianoetico*, in "Materiali per una storia della cultura giuridica", 1986, pp. 489-494.

<sup>69</sup>A mio parere, si può condividere, seppure con talune cautele, la concezione secondo la quale i sistemi normativi in ultima analisi, ascrivono determinate qualifiche deontiche a dei comportamenti.

<sup>70</sup>Sul problema cfr. ROSS A., *TQ-tQ*, in "Scandinavian Studies in Law", I, 1957, pp. 139-153.

prevista una disciplina diversa per il tipo). Alcuni rapporti concettuali possono essere rappresentati mediante regole di produzione: basterà allargare il concetto di regola di produzione giuridica così da ammettere anche regole che connettano ad una determinata condizione non direttamente una conseguenza deontica, ma situazioni e qualificazioni d'altro tipo. Così, ad esempio, potremo avere da un lato, regole del tipo:

se una persona x è nata da genitore italiano,  
allora x è cittadino italiano,

se una persona x è sposata con un cittadino italiano,  
allora x è cittadino italiano<sup>71</sup>,

se una persona x è stata naturalizzata,  
allora x è cittadino italiano

e d'altro lato regole del tipo

se una persona x è cittadino italiano e  
x ha raggiunto la maggiore età  
allora x ha il diritto di voto (art 48 Cost)

se una persona x è cittadino italiano e  
x ha meno di 30 anni e  
x è laureato in giurisprudenza,  
allora  
x può partecipare al concorso per uditore  
giudiziario.

In conclusione, mediante regole di produzione è possibile rappresentare settori del diritto di sufficiente ampiezza da presentare un qualche interesse, solo se comprendiamo tra le regole di produzione giuridiche non solo enunciati immediatamente prescrittivi, che ascrivono obblighi o permessi condizionati (il cui conseguente esprima una qualificazione deontica), ma anche (a) norme di attributive di poteri e, (b) in generale, qualsiasi proposizioni ipotetica giuridicamente significativa che sia sintatticamente qualificabile come regola di produzione (che corrisponda a, o sia riducibile mediante trasformazioni logiche, ad un'implicazione con una congiunzione di antecedenti e un conseguente affermativo).

---

<sup>71</sup>E ricorrono altre condizioni che qui non consideriamo per esteso.

### 3.3. Il diritto è riducibile ad un insieme di regole di produzione?

Il modello delle regole di produzione sembra pertanto adatto ad esprimere un gran numero di contesti giuridici, e in particolare i contesti più semplici, nei quali il diritto determina direttamente, o mediante una qualificazione intermedia, la disciplina di una determinata situazione.

E' più difficile un diretto uso di questo modello per esprimere contesti complessi, nei quali la determinazione della disciplina applicabile non è immediata, ma passa attraverso l'uso di rapporti concettuali, nei quali gioca un ruolo importante l'ereditarietà delle proprietà.

La rappresentazione della conoscenza giuridica in regole di produzione è superficiale, e del tutto insufficiente anche per applicazioni banali, quando le regole di produzione siano formulate al livello della logica enunciativa<sup>72</sup>. Tuttavia, anche rappresentando le regole di produzione al livello della logica dei predicati<sup>73</sup>, questo modello non è del tutto adeguato per un'agevole rappresentazione di contesti giuridici complessi: anche nel mondo del diritto sembra opportuno ricorrere a strutture come reti semantiche o i frames, qualora si debbano considerare rapporti concettuali complessi<sup>74</sup>.

#### 3.3.1 Esempio di regole di produzione giuridiche

Come esempio di regole di produzione giuridiche consideriamo alcune regole del sistema LSD (in materia di responsabilità del produttore)<sup>75</sup>:

---

<sup>72</sup>La maggior parte dei sistemi esperti basati su regole sin qui sviluppati obbediscono a questo semplice modello, ad esempio, il sistema di Oxford, taluni sistemi sviluppati presso l'Istituto di documentazione giuridica di Firenze, il sistema LABEO, il sistema ELI ecc.

<sup>73</sup>Avvicinandosi quindi al modello delle clausole di Horn.

<sup>74</sup>Non è accettabile la tesi di SUSSKIND R.E., *Expert Systems in Law*, p. 151 ss., secondo il quale sarebbe sufficiente l'uso di regole di produzione al livello della logica enunciativa. Solo i casi difficili (*hard cases*) richiederebbero una rappresentazione più approfondita. Come abbiamo già indicato, anche le inferenze più frequenti ed elementari del ragionamento giuridico (ad esempio, il passaggio dalla norma generale al caso concreto) presuppongono almeno la logica predicativa.

<sup>75</sup>Cfr. WATERMAN D.A., PETERSON M.A., *Models of Legal Decisionmaking*, cit., p. 171 ss.

**[REGOLA 1: RESPONSABILITA' PER L'USO DEL PRODOTTO]**

**SE** l'uso di (il prodotto) al tempo del pregiudizio dell'attore è prevedibile  
 e ( quell'uso è ragionevole e corretto  
   o quell'uso è un'emergenza  
   o (vi è una descrizione di quell'uso da parte del convenuto  
     e quella descrizione è scorretta)  
   o non c'è una descrizione di quell'uso da parte del convenuto di quell'uso)  
**ALLORA** afferma che il convenuto è responsabile dell'uso del prodotto.

**[REGOLA7: DILIGENZA COMPARATIVA-RESPONSABILITA' PARZIALE]**

**SE** la teoria della responsabilità stretta (strict liability) non si applica al danno dell'attore  
 e l'utilizzatore del prodotto è la vittima  
 e (California è la giurisdizione della causa  
   o Arizona è la giurisdizione della causa)  
 e la vittima sa che 'il prodotto è pericoloso  
 e l'uso de (il prodotto) da parte dell'utilizzatore è negligente  
 e l'utilizzatore del prodotto contribuisce al danno dell'attore  
**ALLORA** afferma che il convenuto è responsabile del danno dell'attore  
 e la responsabilità del convenuto è parziale  
 e 1 - la proporzione massima della responsabilità per la perdita dell'attore  
   è la proporzione minima della responsabilità del convenuto  
 e 1 - la proporzione minima della responsabilità per la perdita dell'attore  
 è la proporzione massima della responsabilità del convenuto

3.3.2 Regole di produzione e programmazione logica

Come abbiamo già rilevato<sup>76</sup> le regole di produzione (di tipo logico)<sup>77</sup> possono essere facilmente riformulate nella

---

<sup>76</sup>cfr. cap. 7 par. 2.

<sup>77</sup>A queste regole abbiamo fatto esclusivo riferimento nel presente paragrafo.

programmazione logica, spesso ottenendo una formulazione più rigorosa e concisa. In questo caso si ha il vantaggio di integrare le regole (della logica predicativa) in un linguaggio completo, dotato di una sintassi e una semantica definite con precisione, di metodi di inferenza rigorosi e controllabili.

#### 4. Le reti semantiche

Taluni aspetti del diritto sembrano facilmente rappresentabili mediante reti semantiche.

Ad esempio, spesso nel diritto si usano gerarchie concettuali collegate da rapporti di tipo *è un*, realizzate adottando la tecnica della definizione *per genere e differenza specifica*. Questa rappresentazione ha la funzione di semplificare l'esposizione della disciplina giuridica: la normativa comune a più categorie di atti, comportamenti o oggetti è introdotta in sede di disciplina del tipo più generale al quale tali categorie appartengono (ad esempio, il contratto, il reato, il bene mobile ecc.); la disciplina specifica di ogni singola categoria (che integra, e in taluni casi sostituisce, come eccezione, quella generale), è introdotta in sede di disciplina del tipo<sup>78</sup>.

Le classificazioni del legislatore sono completate ed allargate dalla dogmatica giuridica, che introduce a sua volta definizioni dei concetti giuridici (definizioni dottrinali), classifica e sistematizza gli istituti. Così, ad esempio, i fatti giuridici secondo la tradizionale dogmatica privatistica possono essere così classificati come segue:

---

<sup>78</sup>Sulla funzione della disciplina per tipo cfr. TARELLO G., *L'interpretazione della legge*, cit., p. 194 ss.

Fig. 2. La gerarchia dei fatti giuridici



La rete semantica riportata nella figura rappresenta i seguenti rapporti:

fatto giuridico è un fatto,  
 atto giuridico è un fatto giuridico,  
 fatto giuridico In senso stretto è un fatto giuridico;  
 negozio è un atto giuridico;  
 contratto è un negozio;  
 compravendita è un contratto

Come è noto gerarchie concettuali come quella appena presentata sono state oggetto di talune critiche:

-tale tipologia non è conforme al linguaggio nel nostro legislatore (il quale non conosce la categoria del negozio), la costruzione e l'uso di siffatte costruzioni generali (ad esempio il fatto di ricondurre il contratto alla categoria del negozio, o di riferire i diversi contratti alla categoria del contratto) maschera la diversa funzione



sociale dei contratti e di altri negozi, e le differenti funzioni sociali dei diversi tipi contrattuali ecc.<sup>79</sup>.

Nella nostra terminologia potremmo dire che si tratta di un problema di ereditarietà delle proprietà: quali sono le proprietà che noi ascriviamo al concetto di negozio o di contratto, e queste proprietà sono ereditate necessariamente o solo per difetto dai singoli tipi di negozio o di contratto?. Tuttavia questa è solo una risposta parziale: già il fatto di aver costruito questa e non altre gerarchie concettuali ci porta ad impostare i problemi in un certo modo, a cercare determinate analogie, determinate ereditarietà ecc. Qui non approfondiamo la questione, ci importa invece rilevare l'importanza della scelta delle strutture concettuali nelle quali rappresentare la conoscenza giuridica: consentendo di rappresentare più facilmente certi tipi di conoscenza rispetto ad altri, possono esercitare un effetto indiretto sulla formalizzazione del diritto che ne risulta: talune delle questioni che si pongono nell'ambito dell'ingegneria della conoscenza giuridica possono facilmente assumere un rilievo politico-giuridico<sup>80</sup>.

Taluni sistemi basati sulla conoscenza giuridica sono stati sviluppati usando il metodo delle reti semantiche. Ricordiamo, in particolare, il sistema TAXMAN 1 di McCarty usava strutture come quelle riportate nella figura seguente<sup>81</sup>:

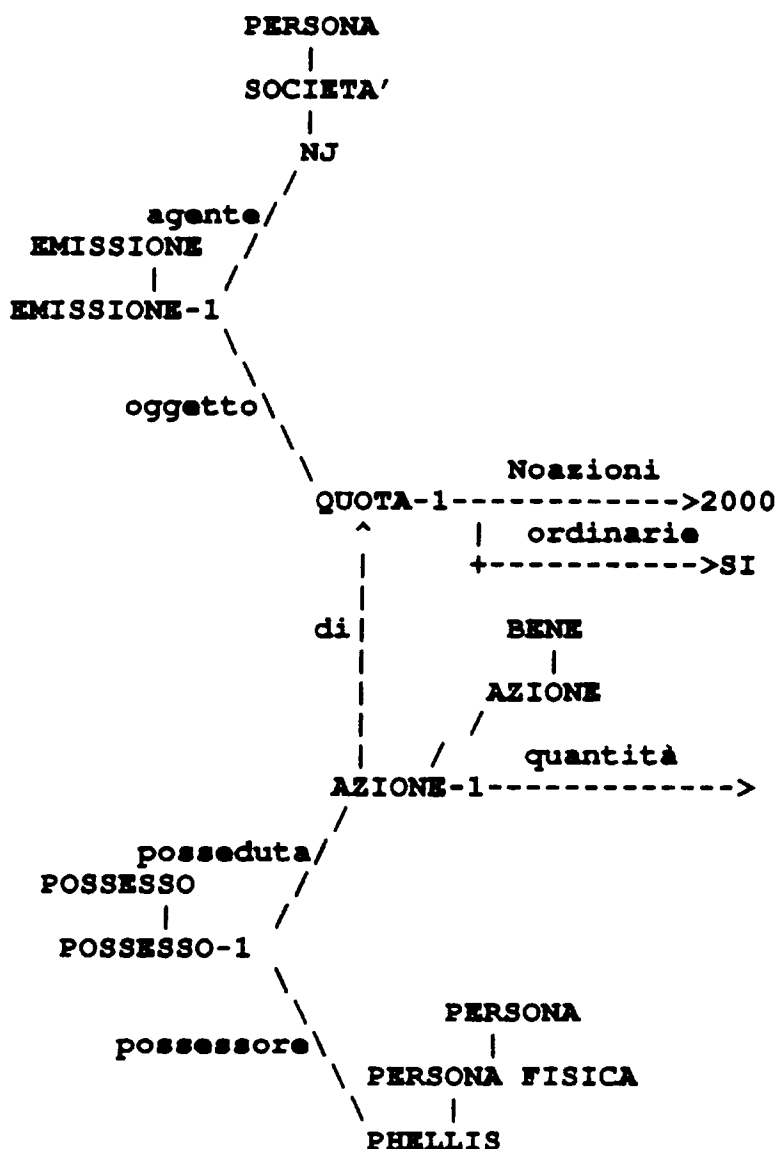
---

<sup>79</sup>Cfr. GALGANO F., voce *Negozio Giuridico* (premesse problematiche e dottrine generali, in *Enc. Dir.* XXVII, 1977, pp. 932 ss.

<sup>80</sup>Si tratta di un problema analogo a quello che può porsi anche per la teoria del diritto (cfr. SCARPELLI U., *La teoria generale del diritto: prospettive per un trattato*, in SCARPELLI U. (a cura di), *La teoria generale del diritto. Problemi e tendenze attuali. Studi dedicati a Norberto Bobbio*, Edizioni di comunità, Milano, 1983, pp. 281-240).

<sup>81</sup>MCCARTY L.T., *The Taxman Project*, cit., p. 26

Fig. 3. Rete semantica di TAXMAN I



(Phellis possiede 100 azioni ordinarie della società New Jersey).

## 5. I frames

Il diritto non si limita ad ascrivere qualifiche deontiche a semplici comportamenti individuali. Esso costituisce e disciplina contenuti concettuali (e realtà sociali) complessi, i cd. istituti giuridici: la persone

giuridica, la società commerciale, il negozio, il contratto, il fatto illecito, il reato ecc.<sup>82</sup>. Questi contenuti concettuali sono spesso inseriti in gerarchie concettuali, nelle quali vi sono forme di ereditarietà delle proprietà<sup>83</sup>.

I *frames* sono strutture particolarmente adeguate per rappresentare contenuti complessi e sviluppare forme di ragionamento per difetto.

Possiamo ad esempio pensare di rappresentare il concetto di contratto con il *frame* seguente:

#### **CONTRATTO**

**è\_un: ATTO GIURIDICO**

**caselle:**

**NUMERO PARTI: (un NUMERO|NUMERO>1 (default:2)**

**CONCLUSIONE: (CONSENSUALE, REALE) (default: REALE)**

**EFFETTI: (OBBLIGATORI, REALI) (default: OBBLIGATORI)**

Che indica che il contratto è un atto giuridico e, di regola (per default) ha due parti, è reale, ha efficacia obbligatoria.

Il linguaggio dei *frames* è stato adottato, ad esempio, nel sistema TAXMAN II, dove la conoscenza è rappresentata con cornici del tipo:

#### **POSSIEDE:**

**è\_un: RELAZIONE**

**caselle:**

**POSSESSORE : un PERSONA**

**POSSEDUTO : un BENE**

**AZIONE:**

**è\_un: OGGETTO**

**caselle:**

**emittente: un SOCIETÀ**

**quantità: un NUMERO**

**valore: un NUMERO**

**con diritto di voto: un SIO NO**

**comune: un SIO NO**

dove con SIO NO indichiamo che la casella deve avere come valore SI o NO (cioè un valore di verità: vero o falso),

*Frames* e programmazione logica non sono approcci incompatibili. Come abbiamo osservato, sono stati proposti

<sup>82</sup>Questo aspetto del pensiero giuridico ha condotto alcuni autori a suggerire un approccio orientato agli oggetti.

<sup>83</sup>Si è considerato questo aspetto nel paragrafo precedente.

<sup>84</sup>MCCARTY L.T., *A Computational Theory of Eisner v. Macomber*, cit. , p. 330.

linguaggi di programmazione logica che comprendono una gestione di *frames*. Il linguaggio per il discorso giuridico di McCarty si ispira ai *frames* per taluni aspetti, pur nell'ambito di un approccio logico.

Alcuni sistemi informatico-giuridici basati sulla programmazione logica comprendono anche una gestione di *frames*. Ad esempio, il sistema PROLEG usa *frames* del tipo seguente per rappresentare contenuti della legge 319 10 maggio 1976 relativa alle norme per la tutela delle acque dall'inquinamento (c.d. legge Merli)<sup>85</sup>.

**Fig. 4. Frame generale in PROLEG**

-----  
**FRAME "scarico generale"**  
 -----

**CARATTERISTICHE: frame generale RIFERIMENTO: nessuno**  
 -----

----- CASELLE -----  
**NOME: Origine VALORI POSSIBILI (pubblico, privato, X)**  
**VALORE DI DEFAULT: X**

**NOME: Tipo VALORI POSSIBILI (diretto, indiretto, X)**  
**VALORE DI DEFAULT: X**

**NOME: Scarico afferente a VALORI POSSIBILI (corso  
 d'acqua superficiale, acque superficiali. acque marine,  
 acque sotterranee pubbliche, X)**  
**Valore di default: X**

**NOME: caratteristiche chimico-fisiche : (XoY)**  
**VALORE DI DEFAULT: X**  
 -----

Il frame riportato qui sotto appartiene alla categoria **scarico generale** (è legato al frame appena presentato da un rapporto di tipo **è\_un**).

---

<sup>85</sup>Cfr. ANDRETTA M., LUGARESI M., ZAMBON F., LOSANO M., NANNINI N., *Uso di linguaggi formali per la rappresentazione di testi normativi: il progetto PROLEG (PROlog applicato alle LEGgi)*, cit., p. 378.

**Fig. 5. Frame speciale in PROLEG**

-----  
**FRAME "scarico"**

-----  
**CARATTERISTICHE: DI TIPO scarico generale RIFERIMENTO:**  
**(legge 319 art. 12 comma 1 a)**

----- CASELLE -----

**NOME: origine VALORI: (privato)**

**NOME: Tipo VALORI: (diretto)**

**NOME: provenienza VALORI: (produttivo)**

**NOME: scarico afferente a VALORI: (Richiesto)**

**NOME: caratteristiche chimico-fisiche VALORI:**  
**(Ricavabile)**  
 -----

## **6. Problemi e limiti della formalizzazione del diritto nella programmazione logica**

Tra gli strumenti per la rappresentazione della conoscenza giuridica si è dedicata un'attenzione particolare alla programmazione logica, cercando di ricondurre ad essa anche altri linguaggi formali. Infatti, in questo volume si è sviluppata la tesi che l'intelligenza artificiale si ispira al modello del sistema deduttivo. La considerazione della programmazione logica, che adotta più direttamente e con maggior rigore questo modello, ci consente di delineare prospettive e limiti di ogni applicazione di intelligenza artificiale.

Le applicazioni giuridiche della programmazione logica, hanno condotto non solo a risultati applicativi determinati<sup>86</sup>, ma soprattutto ad un esame approfondito di problemi, prospettivi e limiti dell'uso di linguaggi formali per rappresentare contenuti giuridici. Sui risultati, sia applicativi che teorici delle esperienze sviluppate presso

---

<sup>86</sup>Nessuna applicazione giuridica dell'intelligenza ha consentito di nonostante la programmazione logica sembri un approccio particolarmente interessante, non si sono ancora raggiunti risultati applicativi significativi.

l'Imperial College, si aperto un interessante dibattito, non privo di punte polemiche.

In questo paragrafo ci si limiterà ad una considerazione di osservazioni interne al paradigma della programmazione logica, che mettono luce i limiti di linguaggi e applicazioni determinate, ma con l'intento di sviluppare soluzioni grazie ad una più raffinata trattazione informatica della logica. Le critiche esterne a quel paradigma, che mettono in discussione la stessa idoneità della logica a rappresentare contesti giuridici, saranno considerate nel capitolo seguente, nell'ambito di una valutazione delle prospettive dell'intelligenza artificiale nel diritto.

Una particolare attenzione sarà dedicata alla programmazione logica in senso stretto. Infatti il linguaggio delle clausole di Horn ha già trovato un'ampia applicazione, mentre le proposte alternative, come quelle sviluppate in LEX (uso dell'intera logica classica) o nel linguaggio per il discorso giuridico di McCarty (uso della logica intuizionistica clausale), seppure riconducibili alla programmazione logica in senso ampio, non hanno ancora trovato applicazioni significative in contesti operativi.

### 6.1. Regole con conclusioni disgiuntive

Nel linguaggio della clausole di Horn ogni regola deve per conclusione un'unico enunciato atomico (o anche una congiunzione di enunciati atomici), non è possibile formulare regole con conclusioni disgiuntive. Il problema del trattamento di regole con conclusioni disgiuntive non sembra pressante a Sergot e al., in quanto, "dopo tutto, la legislazione tende ad essere definitiva". Questa tesi mi sembra fondamentalmente corretta. Una regola con conclusione disgiuntiva stabilisce che, data una certa fattispecie, si verifichi una tra varie conseguenze alternative, senza specificare quale. Le regole che apparentemente presentano questa forma, come ad esempio, le norme che stabiliscono sanzioni alternative<sup>87</sup>, possono essere ricostruite più correttamente come norme che stabiliscono un'unica conseguenza giuridica: esse attribuiscono ad un soggetto determinato il potere di scegliere discrezionalmente una delle alternative<sup>88</sup>.

---

<sup>87</sup>Cfr. ad esempio, art. 594 c.p. (Ingiuria), che recita "Chiunque, offende l'onore o il decoro di una persona presente è punito con la reclusione fino a sei mesi o con la multa fino a lire un milione".

<sup>88</sup>Nell'esempio, il giudice.

Tuttavia, l'uso di regole con conclusioni disgiuntive potrebbe rivelarsi necessario in sistemi che contengano non solo conoscenze giuridiche ma anche conoscenze empiriche. Inoltre, regole con conclusioni disgiuntive sono necessarie per esprimere direttamente condizioni necessarie e sufficienti<sup>89</sup>. Come si vedrà nel seguito regole di questo tipo possono essere formulate anche in modo implicito, grazie alla negazione come fallimento, ma lasciando numerosi problemi irrisolti.

In generale, le regole con conclusione disgiuntiva esprimono una situazione di incertezza: data una certa situazione (descritta dalle condizioni della regola) possono darsi diverse alternative. Le diverse alternative possono essere considerate diverse ipotesi, diverse ricostruzioni alternative, da trattare usando metodi di ragionamento per difetto<sup>90</sup>. In questo modo si potrebbero affrontare anche i casi in cui vi siano diverse possibili interpretazioni.

McCarty, come si è visto, propone di considerare sempre una delle alternative come prototipica, e di trattare le altre come modificazioni o degenerazioni del prototipo. E' dubbio, tuttavia, se in ogni caso in cui si debba scegliere tra più alternative, una sola di queste debba essere considerata il prototipo.

## 6.2. La negazione

Nel linguaggio delle clausole di Horn esteso, le condizioni di una regola possono essere negate, mentre non può essere negata la conseguenza, e la negazione viene interpretata come fallimento (impossibilità di provare, in un tempo finito l'enunciato che si nega). Ciò pone due problemi fondamentali:

a. un problema di capacità espressiva, per quanto attiene all'eventuale esigenza di formulare regole che abbiano per conseguente una negazione;

b. un problema più generale di adeguatezza della negazione come fallimento in contesti giuridici. Questo concetto di negazione è giustificato solo se si muove dall'ipotesi del mondo chiuso<sup>91</sup> cioè dall'ipotesi che la base di conoscenza del sistema comprenda tutta la conoscenza rilevante per decidere il quesito sottoposto al sistema.

---

<sup>89</sup>Il problema si è affrontato illustrando il linguaggio per la rappresentazione della conoscenza giuridica di McCarty.

<sup>90</sup>Cfr. POOLE D., *A Logical Framework for Default Reasoning*, in "Artificial Intelligence", 36, 1988, pp. 27-47.

<sup>91</sup>Cfr. cap. 7 par. 2.

Bisogna quindi considerare se ed in quali limiti questa ipotesi sia accettabile nel diritto.

#### 6.2.1 Conclusioni negative e negazione come fallimento

La negazione come fallimento rende possibile evitare una formulazione esplicita delle regole che stabiliscono conclusioni negative. Infatti, la negazione di un certo enunciato A viene dedotta quando quell'enunciato non sia dimostrabile. Perché il sistema PROLOG risponda negativamente ad un certo quesito  $\neg A$ , data una certa condizione B, basta che, dato B, A non sia deducibile. A tal fine, basterà introdurre la condizione negativa  $\neg B$  in ogni regola che abbia per conclusione A, pur senza introdurre una regola del tipo  $\neg A \leftarrow B$ .

Ad esempio, supponiamo che di aver sviluppato una base di conoscenza in materia di capacità delle persone, e di voler rappresentare, accanto alla regola generale che collega la capacità d'agire al raggiungimento della maggiore età, anche la regola che esclude la capacità per chi sia stato interdetto. Vogliamo che il sistema risponda con un "no" alla domanda "Massimo ha la capacità d'agire?", qualora disponga dell'informazione che Massimo è stato interdetto.

A tal fine basteranno le clausole seguenti:

x è capace d'agire se  
x è maggiorenne e  
 $\neg$  x è stato interdetto.

Massimo è stato interdetto.

Il sistema, al quesito  $\neg$  Massimo è capace d'agire, risponderà no, poiché non sarà possibile soddisfare la condizione  $\neg$  Massimo è stato interdetto.

In generale, le regole che stabiliscono che in un certo caso non vale la qualificazione giuridica prevista da una norma generale debbono essere riformulate come regole che stabiliscono che, in quel caso, sussiste una causa di inapplicabilità della regola generale. La mancanza della causa di inapplicabilità è inserita come condizione aggiuntiva nella regola generale. Se si dimostra la causa di inapplicabilità, allora non sarà più deducibile la conseguenza della norma generale, e il sistema concluderà negando che sussista la qualificazione stabilita dalla norma generale.

Nella formalizzazione del *British Nationality Act* il problema si era presentato per le disposizioni seguenti:



11-(1) ... una persona che subito dopo l'entrata in vigore -

(a) era un cittadino del Regno Unito e delle Colonie, e  
 (b) aveva il diritto di abitare nel Regno Unito a norma dell'Immigration Act del 1971 ..  
 diverrà cittadino inglese all'entrata in vigore

(2) Una persona che ..[P] ..non diventerà cittadino inglese a norma di questa sottosezione (1) a meno che ..[Q]..

dove P e Q indicano certe fattispecie che non saranno specificate nel dettaglio. Questa è la riformulazione proposta da Sergot e al.:

#### Disposizione generale:

x acquista la cittadinanza inglese alla data y a norma della sezione 11.1 se  
 l'entrata in vigore è in y e  
 y1 è immediatamente prima di y e  
 x era un cittadino del Regno Unito e delle Colonie in y e  
 nella data y1, x aveva il diritto di risiedere nel Regno Unito a norma dell'Immigration Act 1971 come allora in vigore e  
 non [la sezione 11.2 impedisce ad x di acquisire la cittadinanza all'entrata in vigore]

#### Eccezione:

la sezione 11.2 impedisce ad x di acquisire la cittadinanza all'entrata in vigore se  
 ...P... e  
 non ...Q...

Questo trattamento delle eccezioni costringe a riformulare tutte le norme speciali come norme derogatorie, e ad inserire in una previsione normativa tutte le deroghe alla disciplina da essa stabilita<sup>92</sup>. Una trattazione più flessibile delle norme derogatorie potrebbe aversi

---

<sup>92</sup>SERGOT M.J., CORY H.T., HAMMOND P., KOWALSKI R.A., KRIWACZEK F., SADRI F., *Formalisation of the British Nationality Act*, cit., p. 379.

ricorrendo a forme di ragionamento per difetto<sup>93</sup>. La conclusione stabilita dalla norma generale vale per difetto, ed è destinata a cadere quando vengano dimostrate le condizioni di inapplicabilità<sup>94</sup>.

### 6.2.2 Ipotesi del mondo chiuso e diritto

In generale, la negazione come fallimento è accettabile solo nella misura in cui sia accettabile l'ipotesi del mondo chiuso. Si considererà il problema dell'accettabilità di questa ipotesi separatamente per i due tipi di conoscenze che un sistema basato sulla conoscenza giuridica può comprendere: conoscenze di diritto, e conoscenze di fatto.

#### a) Ipotesi del mondo chiuso e questioni di diritto

Analizzeremo dapprima il problema rispetto alle conoscenze di diritto.

Si è spesso osservato che una qualificazione giuridica non può che derivare da una norma giuridica. Se nessuna norma stabilisce quella qualificazione - ad esempio, che "Peter è cittadino inglese" - sembra ragionevole che il sistema concluda nel senso che quella qualificazione non sussiste<sup>95</sup>. E' questa la conclusione che il motore

---

<sup>93</sup>La trasformazione delle norme speciali in norme derogatorie e l'inserimento della deroga nella norma derogata, sono imposte anche dall'esigenza di evitare contraddizioni nella base di conoscenza. Questo approccio corrisponde alla tesi filosofica-giuridica che le antinomie non riguardino le norme, il risultato finale dell'interpretazione, ma solo un risultato intermedio del processo interpretativo. In questa prospettiva, la loro soluzione può essere considerata un momento del processo di formalizzazione. Potremmo, pensare, invece a sistemi che, a partire da una base di conoscenza che contiene enunciati logicamente contraddittori, provvedano alla risoluzione delle antinomie in base a criteri determinati. Bisogna però ricordare che vi sono più criteri per risolvere le antinomie (*lex posterior derogat anteriori*, *lex specialis derogat generali*, *lex superior derogat inferiori*), che non c'è un criterio dei criteri, che l'applicazione degli stessi criteri non è priva di incertezza (in particolare per quanto attiene al criterio di specialità).

<sup>94</sup>Sono stati sviluppati sistemi che cercano di realizzare questo trattamento delle eccezioni. In particolare, va menzionato il sistema OBLOG.

<sup>95</sup>Una trattazione adeguata del problema richiederebbe un approfondimento della problematica delle lacune nel diritto. Mi limito ad osservare che la negazione come fallimento mi sembra consentire un trattamento

inferenziale del PROLOG dedurrà, usando il concetto di negazione come fallimento: l'impossibilità di provare l'enunciato "Peter è inglese", condurrà il sistema a negare quell'enunciato.

Ciò significa che la normativa formalizzata nel sistema è considerata come tassativa: nel rappresentare certe norme nel sistema, non ci si limita ad indicare che a certe fattispecie consegue una certa qualificazione giuridica, implicitamente si specifica anche che tale qualificazione consegue solo ad una fattispecie disciplinata espressamente nel sistema<sup>96</sup>.

Non sempre la normativa giuridica rappresentata nella base di conoscenza di un sistema informatico può essere considerata esaustiva:

(a) Gli ordinamenti giuridici delle società moderne sono tanto complessi che ogni formalizzazione di contenuti giuridici corrisponde solo ad un frammento dell'ordinamento. Non sempre si può essere sicuri di aver rappresentato nella base di conoscenza tutte le regole giuridiche rilevanti per il tipo di problemi di cui si occupa il sistema informatico<sup>97</sup>. Il fatto che una determinata conclusione giuridica non sia deducibile dal frammento dell'ordinamento che si è formalizzato non implica che sarebbe impossibile dedurre quella conclusione da una base assiomatica più ampia (sviluppata formalizzando un ulteriore frammento dell'ordinamento)<sup>98</sup>.

(b). Nel ragionamento giuridico si giunge all'affermazione della esistenza (o dell'inesistenza) di una determinata situazione giuridica, non solo grazie ad inferenze deduttive da premesse date, ma spesso anche attraverso argomentazioni nelle quali vengono individuate le premesse idonee a fungere da base assiomatica della

---

"kelseniano" del problema (cfr. KELSEN H., *Reine Rechtslehre*, cit., p. 251 ss.).

<sup>96</sup>La conseguenza giuridica vale se e solo se si verifica una delle ipotesi previste, il se è esplicito, mentre il solo se è implicito. Per una trattazione generale del problema, cfr., KOWALSKI R.A., *Logic for Problem Solving*, cit., p. 210 ss.

<sup>97</sup>I diversi settori del diritto sono collegati tra loro. Ad esempio, la normativa in materia di cittadinanza, richiama quella in materia di famiglia, di attribuzione della personalità, di adozione

<sup>98</sup>Ad esempio, la responsabilità di Massimo in una particolare fattispecie può non essere deducibile in base alle norme generali sulla responsabilità civile, ma tuttavia esservi una norma speciale, in ipotesi non ancora "insegnata" al sistema, in base alla quale Giulio sia responsabile in quella fattispecie. Il sistema, concludendo che Massimo non è responsabile, giungerebbe ad una conclusione falsa,

deduzione logica cogente che conduca all'affermazione di una determinata conclusione. La base assiomatica dalla quale muove un'argomentazione giuridica è tale da poter essere integrata da ulteriori premesse (espressione di valori, principi ecc.) così da fondare sul piano deduttivo (sul piano della così detta giustificazione interna) la conclusione di cui si tratta.

In relazione ai due aspetti ricordati, ogni assiomatizzazione dell'ordinamento giuridico è la formalizzazione di un aspetto parziale e, di conseguenza, non sarebbe corretto assumere che tutta la conoscenza rilevante sia memorizzata in una base di conoscenza.

Tuttavia, l'uso della negazione come fallimento, può risultare accettabile, in certe condizioni, per quanto riguarda le conoscenze di diritto. Infatti, i problemi per il quale il sistema non disponga di una soluzione possono trovare una risposta nell'attivazione di un meccanismo d'interrogazione di un utente esperto (al quale il sistema si rivolga prima di dedurre che non sussiste una certa situazione giuridica), e, più in generale, considerando la base di conoscenza come condivisa tra sistema e utente. Il giudice (e, in un certo senso, ogni giurista) deve comunque prendere una decisione, e tale decisione va adottata a partire dalla premessa che "il giudice sa il diritto". Quindi, l'ipotesi che il diritto sia un mondo chiuso, anche se empiricamente discutibile, sta alla base del funzionamento del nostro ordinamento giuridico. Se né il sistema né l'utente esperto (in particolare, il giudice) sono in grado di indicare un contesto normativo dal quale sia deducibile una certa qualificazione giuridica, può essere ragionevole concludere negando che quella qualificazione sussista

#### b) Ipotesi del mondo chiuso e questioni di fatto

Il problema del trattamento della negazione è più complesso per quanto attiene agli enunciati che descrivono situazioni di fatto. Consideriamo l'esempio seguente:

**P commette reato(R) se**

**P compie fatto(F) e**

**fatto(F) costituisce reato(R) e**

**not causa-giustificazione(G) concorre con fatto(F)**

Data questa regola, il sistema concluderà che Massimo commette il reato di violenza privata, quando sia stato dimostrato che Massimo ha compiuto un fatto costituente violenza privata, e non sia stato dimostrato che concorra con quel fatto una causa di giustificazione. Il sistema concluderà per la colpevolezza anche quando sussista il dubbio sulla presenza della clausola di giustificazione,

mentre in questo caso secondo l'opinione dominante, bisognerebbe assolvere, in base alla regola *in dubio pro reo*. Per rappresentare correttamente la regola bisogna introdurre un concetto di negazione diverso, più vicino alla negazione logica: per dimostrare la negazione di un enunciato negativo non sarà sufficiente l'impossibilità di provare l'enunciato ma occorrerà un'apposita dimostrazione<sup>99</sup>.

Il problema della negazione di enunciati che descrivono situazioni di fatto è discusso da Sergot e al. in relazione alla sezione 1-(2). del *British nationality Act*, che pone gli stessi problemi dell'esempio appena riportato:

**Testo originale**

(2) Un neonato che, dopo l'entrata in vigore, venga trovato abbandonato nel Regno unito, a meno che non si dimostri il contrario, sarà ritenuto, ai fini della sottosezione (1)  
 (a) nato nel Regno Unito dopo l'entrata in vigore, e  
 (b) nato da genitore che al momento della nascita era cittadino britannico o residente nel Regno unito.

**Testo formalizzato:**

x acquista la cittadinanza inglese nella data y a norma della sezione 1.2 se  
x è stato trova come neonato abbandonato nel Regno Unito e  
x è stato trovato nella data y e  
y è precedente o successiva all'entrata in vigore  
e  
not [x non è nato nel Regno unito dopo l'entrata in vigore] e  
not [x non è nato da un genitore qualificato a norma della sezione 1.1 al momento della nascita].

Sergot e al. osservano che, per cogliere il senso di questa disposizione, bisogna interpretare la prima negazione come impossibilità di prova (negazione come fallimento) e la

---

<sup>99</sup>Aver individuato questa esigenza non significa aver trovato una soluzione. Il problema del trattamento della negazione è il problema fondamentale della programmazione logica. Finora sono state sviluppate numerose proposte ma non è stata trovata alcuna soluzione del tutto soddisfacente. In questa sede non è possibile approfondire il problema.

seconda come negazione classica. Si consideri, ad esempio, l'enunciato

not [x non è nato da un genitore qualificato a norma della sezione 1.1 al momento della nascita].

Se le due negazioni si interpretano entrambe come negazioni classiche (è falso che sia falso x è nato da un genitore qualificato)<sup>100</sup> o come fallimento (non è provabile che non è provabile che x è nato ...) otteniamo enunciati equivalenti a "x è nato da un genitore qualificato a norma della sezione 1.1 al momento della nascita". Nell'impossibilità di provare questo enunciato, non è a possibile concludere che x è cittadino inglese (come invece la norma prescrive).

A mio parere, l'impossibilità di formalizzare le norme appena considerate nel linguaggio delle clausole di Horn estese con la negazione come fallimento, dipende da quanto segue. Non solo il contesto delle conoscenze fattuali su cui si basano le decisioni giuridiche è un mondo aperto nella realtà (ciò vale spesso anche per il contesto delle conoscenze normative), ma lo è anche nella considerazione che ne fa il diritto. Il diritto è consapevole la decisione giuridica si pone in un mondo aperto: niente garantisce che tutti i fatti rilevanti siano pienamente provati e, anche in mancanza della prova di tutti i fatti deve comunque essere presa una decisione.

Pertanto, il diritto disciplina non solo i fatti, ma anche la loro prova e determina, di volta in volta, le conseguenze della mancata prova di elementi di fatto. Talvolta la mancanza della prova di un certo enunciato equivale alla prova alla negazione dell'enunciato stesso, altre volte essa equivale alla prova dell'enunciato, altre volte è soggetta ad una disciplina ancora diversa. Come è noto, nell'insieme delle condizioni da cui dipende un effetto giuridico, si distingue infatti tra i c.d. fatti costitutivi<sup>101</sup>, che debbono essere provati (non basta che non sia dimostrabile il fatto contraddittorio) affinché il giudice debba decidere che si è verificata quell'effetto giuridico e altri fatti (collaterali, successivi o occasionali) si assumono a condizione che non venga provato

---

<sup>100</sup>Dove P significa "X è nato nel regno Unito dopo l'entrata in vigore"

<sup>101</sup>Si tratterebbe dei fatti principali, di quelli che, in generale, a prescindere da particolari circostanze collaterali o successive, sono idonei a causare l'effetto giuridico di cui si tratta. Talvolta è la legge stessa che, invertendo l'onere della prova, qualifica un certo fatto come costitutivo o il fatto contraddittorio come fatto impeditivo o estintivo.

il fatto contraddittorio<sup>102</sup>. In generale, i fatti costitutivi sono positivi e i fatti collaterali sono fatti negativi, ma vi sono anche fatti collaterali positivi e fatti principali negativi<sup>103</sup>. La negazione di un fatto collaterale (o il fatto contraddittorio) che è chiamata fatto impeditivo, modificativo o estintivo del fatto principale<sup>104</sup>. La distinzione dei fatti in costitutivi e secondari, è operata dall'interprete, o anche da specifiche norme, le c.d. presunzioni relative, che trasformano un fatto costitutivo A in un fatto secondario, cioè, in altri termini, che privano il fatto A della qualifica di fatto costitutivo ed elevano il fatto contraddittorio non-A a fatto modificativo o estintivo. La presunzione relativa modifica la strategia di prova: ciò che importa per desumere l'effetto C, non è provare la condizione A, ma fallire la prova di non-A<sup>105</sup>.

---

<sup>102</sup>La mancanza di uno di quei fatti, rappresenta un fatto impeditivo o estintivo, che spetta al convenuto provare affinché sia respinta la domanda dell'attore che ha provato i fatti costitutivi di un certo diritto.

<sup>103</sup>Ad esempio il diritto del compratore alla consegna della cosa dipende sia dall'esistenza di un contratto di compravendita, sia dalla titolarità giuridica, in capo al venditore, della cosa venduta. Affinché il giudice debba concludere nel senso il compratore ha diritto alla consegna della cosa, è necessario che venga provato il primo fatto, mentre non è necessario che venga provato il secondo (basta che non venga provato il fatto contraddittorio, cioè che il venditore non è titolare della cosa). Se formalizzassimo la norma come:

**il compratore x ha diritto alla consegna della cosa y da parte del venditore z se**

**z ha venduto y a x e**

**z era proprietario di y alla data della vendita.**

Il sistema dedurrebbe che X non ha diritto alla consegna della cosa ogni volta che non si provi che Z era proprietario (a prescindere dal fatto che si provi che egli non era proprietario).

<sup>104</sup>Ad esempio, il fatto che il venditore non abbia la titolarità giuridica del bene, il fatto che chi contrae matrimonio non abbia l'età richiesta ecc.

<sup>105</sup>Ad esempio, a norma degli artt. 1153 e 1147 cc., chi abbia acquistato un bene mobile da chi non ne era proprietario deve essere considerato il proprietario del bene, a meno che non si dimostri che egli non era in buona fede. Non è necessaria la prova del fatto positivo (che l'acquirente era in buona fede), ma basta che fallisca la prova del fatto negativo (che l'acquirente non era in buona fede).

Ma ciò significa che **non-A**, un enunciato negativo deve divenire il goal da dimostrare<sup>106</sup>, e che tale enunciato non può essere equivalente a **n-A**, dove **n-** è la negazione come fallimento<sup>107</sup>. Se si accetta l'ipotesi del mondo chiuso, la normativa in materia di presunzioni diventa irrilevante, e il fatto **A** sul quale verte la presunzione relativa può essere espresso con una delle due formulazioni equivalenti **non non A** e **n-n-A** (dove i simboli **n-** rappresentano negazioni come fallimento). Diventa inutile rappresentare norme come quella sopra menzionata.

Sergot e al. risolvono questo caso, trattando l'enunciato contenente la negazione classica come un enunciato atomico: il **non** viene inglobato in un nuovo predicato usato per interrogare l'utente<sup>108</sup>.

Questa soluzione non sembra soddisfacente in generale. Si tratta solo di un modo di aggirare il problema: il significato della negazione classica dipende non dall'uso che ne faccia il motore inferenziale, ma solo dal significato che l'utente attribuisce al predicato che contiene la negazione, o da regole ad hoc.

Kowalski stesso<sup>109</sup> suggerisce una diversa soluzione: si tratta di interpretare la norma che stabilisce una presunzione come un enunciato metalinguistico, che afferma che i fatti oggetto della presunzione relativa valgono per abduzione qualora siano dimostrati i restanti elementi della fattispecie (i fatti costitutivi).

In questo modo potrebbe essere trattata, più in generale, la distinzione tra fatti costitutivi e restanti elementi della fattispecie. Questa soluzione pone però difficili problemi di tipo informatico: occorre sviluppare metodi di ragionamento per difetto (i fatti presunti debbono

<sup>106</sup>Per concludere, eventualmente, che il fatto non è dimostrabile.

<sup>107</sup>Se così fosse, dovremmo concludere (interpretando entrambe le negazioni **n-** e **non** come negazioni come fallimento) che la prova di **n-non-A** equivale alla prova di **A**.

<sup>108</sup>Viene introdotto il predicato

non è nato nel Regno unito prima dell'entrata in vigore.

Il sistema chiede all'utente se Peter soddisfa questo predicato, proponendo il seguente enunciato:

Peter non è nato nel Regno unito prima dell'entrata in vigore,

Se la risposta è negativa, il sistema non è in grado di dimostrare l'enunciato e ne deduce (negazione come fallimento) che

not [Peter non è nato nel Regno unito prima dell'entrata in vigore],  
realizzando la condizione della regola.

<sup>109</sup>KOWALSKI R.A., *Notes on the British Nationality Act*, non pubblicato, 1988, p. 36.



cadere se si dimostra il contrario) e, comunque, non basta la negazione come fallimento (i fatti presunti possono essere positivi o negativi). Inoltre, i fatti di cui si tratta non debbono essere assunti in generale, ma solo ai fini dell'applicazione della norma di cui si tratta.

c) Conclusione. Negazione nella programmazione logica e nel diritto.

In conclusione, la negazione come fallimento non è sufficiente a trattare contesti giuridici. Infatti, questo tipo di negazione è adeguato nei limiti in cui sia accettabile l'ipotesi del mondo chiuso.

Questa ipotesi può essere forse a per quanto attiene alle conoscenze di diritto, qualora la base di conoscenza si consideri condivisa tra sistema ed esperto (anche se, la c.d. presunzione di conoscenza del diritto si rivela sempre più una finzione, anche per l'esperto).

Invece, l'ipotesi del mondo chiuso è inaccettabile per le conoscenze di fatto. Infatti, il diritto prende in considerazione implicitamente o esplicitamente non solo i fatti, ma anche la loro prova. La prova è disciplinata in vario modo, e non necessariamente la mancanza della prova di un enunciato equiparata alla prova della falsità di quell'enunciato. Per quanto riguarda le conoscenze di fatto, l'assunto del mondo chiuso è non solo irrealistico, ma incompatibile con l'esigenza di rappresentare fedelmente il diritto e il contesto della decisione giuridica.

Ciò non significa che sia inutilizzabile nel diritto il modello della programmazione logica in senso stretto: non esistono oggi strumenti informatici in grado di trattare adeguatamente il discorso giuridico, e nell'analisi comparativa degli strumenti disponibili, il metodo della programmazione logica si presenta, complessivamente, come il più adeguato. La soluzione deve ricercarsi non in un'alternativa, ma in un'estensione del linguaggio delle clausole di Horn definite, estensione che comprenda, in particolare: (a) l'uso di un altro tipo di negazione (ad esempio, la negazione classica o quella intuizionista) accanto alla negazione come fallimento (b) forme di ragionamento per difetto. Soluzione di questo tipo, come abbiamo visto, sono suggerite sia da McCarty che da Kowalski e Sergot<sup>110</sup>.

---

<sup>110</sup>Cfr., su questo problema, GABBAY D.M., SERGOT M.J., *Negation as Inconsistency. I*, in "Journal of Logic Programming", 1986, 1, pp. 1-35.

### 6.3. I condizionali soggettivi

Sergot et al. osservano che l'uso delle clausole di Horn pone taluni difficili problemi per i condizionali contraffattuali<sup>111</sup>.

Forse il problema è di carattere più generale. Il trattamento dell'implicazione nella logica e il rapporto con il linguaggio naturale, è uno dei problemi più controversi della logica moderna. I condizionali soggettivi, che sono enunciati che parlano di ciò che accadrebbe (o sarebbe accaduto) se le cose andassero (o fossero andate) in un certo modo, non corrispondono alla comune implicazione materiale della logica: essi esprimono rapporti causali o altri tipi di condizionamento. Consideriamo, ad esempio, il seguente enunciato:

**SE la persona x non conclude il contratto,  
ALLORA la persona x subisce un grave danno alla persona.**

Interpretando l'implicazione come la comune implicazione materiale dovremmo considerare questo enunciato dovrebbe considerarsi equivalente al seguente:

**la persona x conclude il contratto, OPPURE  
la persona x subisce un grave danno.**

Quest'ultimo enunciato risulta vero in ogni caso in cui la persona di cui si tratta conclude il contratto. Invece il senso dell'enunciato richiede che esso è vero se e solo se sussista un rapporto tra mancata conclusione e danno, a prescindere dal fatto che il contratto sia stato concluso o meno. Infatti, si tratta di un condizionale soggettivo, che specifica ciò che accadrebbe nell'ipotesi la x non concluda il contratto.

Questo enunciato può essere incluso in un enunciato più complesso, come il seguente, che intende esprimere il concetto di stato di pericolo ai sensi dell'art. 1447 cc.

**La persona x si trova in stato di pericolo SE  
[SE [non [la persona x conclude il contratto]  
ALLORA [la persona x subisce un grave danno alla  
persona]].**

La traduzione in una clausola di Horn darebbe il seguente risultato:

---

<sup>111</sup>SERGOT M.J., CORY H.T., HAMMOND P., KOWALSKI R.A., KRIWACZEK F., SADRI F., Formalisation of the British Nationality Act, cit., p. 382.

La persona x si trova in stato di pericolo SE  
 [la persona x conclude il contratto]  
 O [la persona x subisce un grave danno alla  
 persona]].

Che corrisponde alle due clausole seguenti:

La persona x si trova in stato di pericolo SE  
 [la persona x subisce un grave danno alla  
 persona].

La persona x si trova in stato di pericolo SE  
 [la persona x conclude il contratto].

nessuna delle quali esprime fedelmente il senso della norma.

La soluzione suggerita da Sergot e al. consiste nel tradurre il condizionale soggettivo in una disgiunzione di enunciati atomici, che rappresentino le possibili situazioni nelle quali il condizionale è vero. Si tratta di una soluzione che è insoddisfacente come soluzione di carattere generale<sup>112</sup>, in quanto nella maggior parte dei casi sembra impossibile determinare in astratto tutte quelle situazioni. Tuttavia, vi sono interessanti prospettive per la soluzione di questo problema pur restando nell'ambito di un approccio ispirato alla logica. In particolare, Sergot et al. suggeriscono di usare un linguaggio che amalgami linguaggio e metalinguaggio, così da poter esprimere il rapporto di condizionamento tra una certa ipotesi si aggiunge alla base di conoscenza (la conclusione del contratto) ed un certo fatto (il danno)<sup>113</sup>. McCarty propone invece di rappresentare i condizionali soggettivi mediante clausole di Horn annidate (la condizione di una clausola di Horn può essere, a sua volta, una clausola di Horn) interpretate nella logica intuizionistica.

#### 6.4. La considerazione dei tipi

La formalizzazione del *British Nationality Act* tratta oggetti di tipo diverso senza considerare il tipo degli stessi. Così, ad esempio, nella clausola

---

<sup>112</sup>Come riconoscono anche SERGOT M.J., CORY H.T., HAMMOND P., KOWALSKI R.A., KRIWACZEK F., SADRI F., *Formalisation of the British Nationality Act*, cit., p. 382.

<sup>113</sup>Ivi, p. 382.

x acquista la cittadinanza inglese alla data y a norma della sezione 11.1 se  
l'entrata in vigore è in y e  
y<sup>1</sup> è immediatamente prima di y e  
x era un cittadino del Regno Unito e delle Colonie in y e  
nella data y<sup>1</sup>, x aveva il diritto di risiedere nel Regno unito a norma dell'Immigration Act 1971 come allora in vigore e  
not [la sezione 11.2 impedisce ad x di acquisire la cittadinanza all'entrata in vigore]

non è specificato formalmente che x è una persona e che y e y<sup>1</sup> sono date. Ciò risulta solamente dalla parola data compresa nel predicato. Tuttavia, nella logica predicativa del primo ordine, l'intero predicato rappresenta un'unità inscindibile, esso è trattato come una sequenza di caratteri arbitrari, e quindi, in questo senso, il programma non "sa" che i quelle espressioni designano oggetti di un tipo determinato<sup>114</sup>: potrebbe concludere che il cane Fido è un cittadino inglese, o che Fido è nato nella data "Antonio". Una soluzione al problema dei tipi può aversi specificando il tipo con un predicato aggiuntivo.

x acquista la cittadinanza inglese alla data y a norma della sezione 11.1 se  
x è una persona e  
y è una data e  
l'entrata in vigore è in y e  
y<sup>1</sup> è immediatamente prima di y e  
y<sup>1</sup> è una data e  
x era un cittadino del Regno Unito e delle Colonie in y e  
nella data y<sup>1</sup>, x aveva il diritto di risiedere nel Regno unito a norma dell'Immigration Act 1971 come allora in vigore e  
non [la sezione 11.2 impedisce ad x di acquisire la cittadinanza all'entrata in vigore] e  
11.2 è una sezione.

Questa formulazione rende la rappresentazione formale della disciplina giuridica prolissa e controintuitiva e può presentare notevoli alcuni problemi quando ci si proponga approfondire il trattamento del linguaggio naturale.

---

<sup>114</sup>Si è già rilevato che un sistema automatico è in grado di capire solo ciò che sia espresso in precise strutture formali.

McCarty<sup>115</sup> suggerisce di adottare una *sorted logic*, nella quale i predicati che definiscono i tipi vengono trasformati in etichette di tipo sulle variabili e le costanti. Le etichette di tipo possono essere collegate in reti semantiche, in modo da che risulta più facile la gestione di aspetti come l'ereditarietà delle proprietà, e il rapporto tra i tipi può essere gestito direttamente dal motore inferenziale<sup>116</sup>.

Una soluzione forse più flessibile (anche se meno efficiente) consiste nel trattare i vincoli sui tipi come vincoli di integrità (*integrity constraints*) del programma logico.

### 6.5. I concetti intensionali

Il linguaggio naturale, nel quale sono espressi i testi giuridici, comprende numerosi concetti intensionali, cioè espressioni che hanno per argomento interi enunciati, che si riferiscono a stati di cose o a contenuti concettuali, e non ad individui in senso stretto. Pensiamo a verbi come "pensare che", "credere che", "sapere che", "affermare che", "chiedere se" ecc., ai verbi modali ("dovere", "potere" ecc.), alle determinazioni temporali.

Un linguaggio formale per il diritto dovrebbe comprendere, in generale la possibilità di trattare concetti intensionali, non solo nei casi comunemente ricompresi nella logica modale o deontica. Pensiamo ad esempio alla fattispecie della calunnia<sup>117</sup>, che coinvolge situazioni epistemiche complesse:

**Una persona commette il reato di calunnia se una persona P1 afferma che una persona P2 è colpevole di un fatto F1 costituente reato, ma P1 sa che P2 non è colpevole.**

Nelle clausole di Horn e, più in generale, nella logica predicativa del primo ordine abbiamo a disposizione due soli

---

<sup>115</sup>Cfr. cap. par. 2.3.1.

<sup>116</sup>Cfr. AIT-KACI H., NASR R., *Login: A Logic Programming Language with Built-in Inheritance*, in "Journal of Logic Programming", 3, 1986, pp. 185-205.

<sup>117</sup>Art. 368 c.p. (Calunnia). "Chiunque, con denuncia, querela, richiesta o istanza, anche se anonima o sotto falso nome, diretta all'Autorità giudiziaria o ad un'altra Autorità che a quella abbia l'obbligo di riferirne, incolpa di un reato taluno che egli sa essere innocente, ovvero simula a carico di lui le tracce di un reato, è punito con la reclusione da due a sei anni".

ingredienti per rappresentare gli enunciati atomici: predicati, che esprimono relazioni e termini, che designano individui. Per ottenere formulazioni sufficientemente astratte siamo costretti a considerare anche le relazioni, gli stati di cose come individui. Altrimenti, la fattispecie generale la fattispecie generale della calunnia dovrebbe scindersi in numerose fattispecie, ciascuna delle quali riguardi un determinato reato. Ad esempio:

P<sup>1</sup> commette il reato di calunnia avente ad oggetto il furto dell'oggetto O nella data D se  
P<sup>1</sup> accusa P<sup>2</sup> di aver commesso il furto  
dell'oggetto O nella data D e  
P<sup>1</sup> sa che P<sup>2</sup> non ha commesso il furto  
dell'oggetto O nella data D,

Questa rappresentazione è superficiale, in quanto un predicato come (1) **commette il reato di calunnia avente ad oggetto il furto dell'oggetto** (2) **nella data** (3) rappresenta solamente una sequenza arbitraria di caratteri. Il sistema non capisce che si tratta di un concetto composto di una combinazione di concetti elementari più semplici (il concetto di commettere, che ha per oggetto il concetto di reato, qualificato nei termini del concetto di calunnia ecc.).

Invece, potremmo riformulare l'enunciato summenzionato come segue:

**commettere(x, reato(calunnia(x,y)) se**  
**affermare(x, (passato(commettere(y, reato(R)))) e**  
**sapere(x, (non(passato(commettere(y, reato(R))))).**

considerando anche le relazioni come individui da designare con nomi complessi (con funzioni)<sup>118</sup>.

Una sintassi di questo tipo può essere adottata anche nel PROLOG. Aver definito una sintassi che consente di esprimere certi concetti non significa, purtroppo, aver sviluppato un trattamento logico di questi concetti. Una logica per la rappresentazione della conoscenza giuridica dovrebbe comprendere logiche modali, deontiche, epistemiche, temporali dell'azione e ogni altra logica necessaria per affrontare i concetti del linguaggio comune. Infatti, come abbiamo osservato, il discorso giuridico è profondamente immerso nel linguaggio naturale e, a differenza dei linguaggi scientifici ne conserva tutta la complessità.

---

<sup>118</sup>Per quanto attiene alla possibilità di trattare anche stati di cose come individui, ricordo che sia nel sistema LEX che nel linguaggio proposto da McCarty eventi o relazioni vengono trattati come individui.

Il problema di un trattamento logico dei concetti intensionali non ha trovato sinora una soluzione soddisfacente in generale<sup>119</sup>. Una soluzione provvisoria può consistere nell'adottare una rappresentazione "profonda" dei concetti intensionali, senza sviluppare uno speciale trattamento logico.

#### 6.6. L'interpretazione del fatto

Nella formalizzazione del *British Nationality Act* si è scelta la soluzione più semplice del problema della interpretazione del fatto<sup>120</sup>, cioè l'uso diretto del linguaggio legislativo. Come osservano Sergot e al. sarebbe possibile ricorrere anche all'approccio consistente nell'introduzione di ulteriori definizioni, per ridurre il linguaggio della legge ad un linguaggio più semplice e vicino all'utente. Lo sviluppo di un dialogo in linguaggio naturale postula, invece lo sviluppo di un modello concettuale profondo del settore del diritto analizzato.

#### 6.7. Conclusione

L'uso di metodi di programmazione logica per rappresentare la conoscenza giuridica è ancora agli inizi. Vi sono numerosi problemi ancora da risolvere (alcuni dei quali si sono illustrati nelle pagine precedenti). Tuttavia, nessun altro approccio ha consentito di ottenere risultati tanto promettenti e, come si è visto, i risultati ottenibili con altri metodi possono essere riformulati nel paradigma della programmazione logica. Nei paragrafi precedenti abbiamo indicato alcuni dei problemi di un approccio logico limitato alla programmazione logica e/o alla logica dei predicati. Siamo rimasti, tuttavia, all'interno di un paradigma logico-deduttivo.

La maggior parte delle critiche rivolte alla programmazione logica nel diritto non vertono tanto sull'adeguatezza di un determinato linguaggio logico o di una determinata metodologia per la rappresentazione della conoscenza rispetto all'obiettivo di una rappresentazione

---

<sup>119</sup>Vi sono infiniti problemi teorici irrisolti nelle logiche che vanno al di là della logica classica dei predicati. A questi si aggiungono i problemi di un'efficiente implementazione informatica. Sulle logiche non standard nell'intelligenza artificiale, cfr. TURNER R., *Logics for Artificial Intelligence*, Ellis Norwood, Chichester (England), 1984.

<sup>120</sup>Sul problema dell'interpretazione del fatto, cfr. cap. 6 par. 4.4.

logica del testo legislativo, ma, piuttosto su questo stesso obiettivo.

Queste critiche saranno considerate nel capitolo seguente. Infatti, un paradigma logicistico caratterizza non solo la programmazione logica, ma tutte le applicazioni di intelligenza artificiale (la programmazione logica ha solo il merito di aver applicato quel paradigma nel modo più rigoroso e completo). Pertanto, le critiche all'uso di metodi di tipo logico in generale vanno lette come critiche alla possibilità di una rappresentazione formale della conoscenza giuridica in generale e quindi, come obiezioni rivolte contro ogni applicazione informatico-giuridica intelligente (almeno allo stato della tecnica).



1. La logica nel diritto (cenni)	200
1.1. La logica enunciativa. La normalizzazione di L.E. Allen	203
1.1.1 Primo esempio : la normalizzazione della normativa sulla capacità di contrarre matrimonio	206
a) Il requisito dell'età	206
b) Le condizioni del potere di contrarre matrimonio	207
1.1.2 Secondo esempio, la normalizzazione di un articolo della costituzione italiana	207
a) Testo originale	207
b) Formula logica	207
c) Testo normalizzato	208
d) Schema del testo normalizzato	208
Diagramma a freccia	209
1.1.3 Il significato della normalizzazione	209
1.2. La logica predicativa	212
1.2.1 Formalizzazione di un principio generale del diritto	212
a) Principio	212
b) Glossario:	213
c) Formalizzazione nella logica predicativa	213
1.2.2 Formalizzazione di un'argomentazione giudiziaria	213
a) Fatti della causa:	213
b) Testo dell'argomento della difesa:	213
c) Struttura logica dell'argomento:	214
d) Glossario:	214
e) Premesse dell'argomento:	214
f) Conclusione	215
g) Deduzione	215
2. La programmazione logica	216
2.1. La programmazione logica in senso stretto	216
2.1.1 La formalizzazione del British Nationality Act 17	
a) Testo originale:	218
b) Rappresentazione in clausole di Horn.	218
2.2. La teoria generale del discorso	220
2.2.1 Conoscenze del senso comune	223
2.2.2 Conoscenze giuridiche.	223
2.3. Il linguaggio per il discorso giuridico di McCarty	225
2.3.1 La sintassi delle formule atomiche	226
2.3.2 La sintassi delle formule molecolari	228
a) Le clausole di Horn	228
b) Negazioni e implicazioni annidate	228
c) Il ragionamento per difetto	229
d) Prototipi e deformazioni	230
2.3.3 Le modalità	231
a) Il tempo	231

b) Eventi ed Azioni	232
c) I concetti deontici	232
d) Conclusione	233
3. Le regole di produzione	234
3.1. Le norme ipotetiche	234
3.2. Il diritto è riducibile ad un insieme di obblighi condizionati?	237
3.3. Il diritto è riducibile ad un insieme di regole di produzione?	240
3.3.1 Esempio di regole di produzione giuridiche	240
3.3.2 Regole di produzione e programmazione logica	241
4. Le reti semantiche	242
5. I frames	245
6. Problemi e limiti della formalizzazione del diritto nella programmazione logica	248
6.1. Regole con conclusioni disgiuntive	249
6.2. La negazione	250
6.2.1 Conclusioni negative e negazione come fallimento	251
6.2.2 Ipotesi del mondo chiuso e diritto	253
a) Ipotesi del mondo chiuso e questioni di diritto	253
b) Ipotesi del mondo chiuso e questioni di fatto	255
c) Conclusione. Negazione nella programmazione logica e nel diritto.	260
6.3. I condizionali soggettivi	261
6.4. La considerazione dei tipi	262
6.5. I concetti intensionali	264
6.6. L'interpretazione del fatto	266
6.7. Conclusione	266

## CAP. 9.

# CONCLUSIONE: INTELLIGENZA ARTIFICIALE, LOGICA E RAPPRESENTAZIONE DELLA CONOSCENZA GIURIDICA

Nella prima parte del presente volume si è argomentata la tesi che l'intelligenza artificiale (allo stato della tecnologia) è fondamentalmente ispirata dal paradigma logico del sistema assiomatico-formale. Pertanto, le prospettive e i limiti dell'intelligenza artificiale nel diritto sono state correlate alle prospettive e i limiti di una assiomatizzazione del diritto.

Nella seconda parte si sono illustrati alcuni dei principali approcci al problema della rappresentazione della conoscenza giuridica, ponendo un particolare accento sulla programmazione logica, rapportando a questa i diversi linguaggi e metodi per la rappresentazione della conoscenza.

In queste pagine conclusive si riconsidererà il rapporto tra logica formale ed applicazioni giuridiche dell'intelligenza artificiale:

-Dapprima si esamineranno in generale alcuni aspetti del dibattito sull'uso della logica nell'intelligenza artificiale.

-Quindi si valuteranno alcune critiche dell'uso di un approccio logico nelle applicazioni giuridiche dell'intelligenza artificiale.

-Infine, si ricapitoleranno alcune delle principali conclusioni cui siamo giunti nel presente volume.

## 1. Intelligenza artificiale e logica

L'uso della logica nell'intelligenza artificiale è stato al centro di una viva discussione, il c.d. dibattito pulito/sporco (*neat/scuffy debate*), nel quale sono emersi due distinti approcci all'intelligenza artificiale<sup>1</sup>.

L'approccio c.d. "pulito" consiste nel "preparare programmi per elaboratore che ragionino in base a linguaggi ben stabiliti della logica matematica, a prescindere dal fatto che sia questo il modo nel quale la gente pensa".

L'approccio c.d. "sporco", cerca invece "di far sì che gli elaboratori imitino il modo nel quale la mente umana

---

<sup>1</sup>Cfr. KOLATA G., *How can Computers get Common Sense*, in "Science", 217, 1987, pp. 1237-1238.

funziona ... che sicuramente non è quello della logica matematica"<sup>2</sup>.

L'approccio pulito si ispira alla logica e alla matematica, mira ad una validità formale; l'approccio sporco si ispira al ragionamento comune, mira invece ad una validità psicologica e pragmatica.

Il primo si connette ad un concetto c.d. "debole" di intelligenza artificiale: questa disciplina deve proporsi di "simulare" comportamenti intelligenti, non di riprodurre esattamente i processi psicologici che caratterizzano l'intelligenza umana; il secondo si connette ad un concetto c.d. "forte" di intelligenza artificiale: questa disciplina ha lo scopo di "duplicare" la mente umana, di realizzare sistemi che pensino come gli uomini<sup>3</sup>.

Il primo si collega all'approccio di tipo generalista illustrato nel cap. 3 par. 3.1. e riprende tecniche della logica matematica; il secondo intende far valere le specificità delle diverse intelligenze, i diversi modi nei quali l'intelligenza si atteggia nei diversi settori, e si collega all'approccio più pragmatico illustrato nel cap. 3 par. 3.2.

Entrambe le direzioni di ricerca possono vantare di successi: alcuni programmi funzionanti sono stati realizzati nell'ambito dell'approccio sporco, ma, a mano a mano che si accumulano le ricerche, emergono teorie in grado di consolidare e generalizzare i risultati sinora raggiunti.

Il dibattito pulito/sporco si connette alla contrapposizione procedurale/dichiarativo: le procedure possono fornire una soluzione "sporca" ad hoc per qualsiasi problema, l'approccio dichiarativo si connette all'esigenza di formulare con precisione (in un insieme di assiomi) teorie generali.

A mio parere, proprio nel campo dei problemi difficili, caratteristici dell'intelligenza artificiale, non c'è una vera alternativa ad un approccio pulito-dichiarativo. Infatti, si possono dare diverse risposte al problema se la nostra mente sia un agglomerato di approcci ad hoc o sia un insieme di teorie tendenzialmente generali e coerenti, se il suo funzionamento corrisponda al modello procedurale o a quello dichiarativo. E' indubbio, tuttavia, che l'uomo non può sviluppare e controllare un sistema complesso senza una teoria adeguata, senza astrarre le conoscenze dalla loro

---

<sup>2</sup>Tra i propugnatori di questo approccio vi sono alcuni illustri studiosi, come ad esempio, M. Minski e R. Schank.

<sup>3</sup>Per questa distinzione, cfr. SEARLE J.R., *Minds, Brains and Science*, cit., p. 37.

manipolazione<sup>4</sup>. In questa prospettiva si giustifica la nostra tesi che sistemi informatici intelligenti, sistemi complessi, destinati a fronteggiare problemi difficili, si ispirano fondamentalmente al modello assiomatico.

In particolare, nelle applicazioni giuridiche è preferibile ricorrere ad un approccio pulito-dichiarativo, sviluppare sistemi informatici che funzionino il più possibile in base a precisi e controllabili modelli logico-matematici, piuttosto che in base ad analogie spesso forvianti con la psicologia umana. Infatti, i sistemi c.d. intelligenti non dispongono della capacità di elaborare criticamente gli schemi concettuali alla luce del contesto, l'interazione dell'uomo con i suoi simili e con l'ambiente non può essere sostituita dalle manipolazioni simboliche di cui l'elaboratore è capace: questi sistemi sono affidabili solo nella misura in cui se ne conosca esattamente il comportamento<sup>5</sup>. Inoltre, ragioni non solo informatiche, ma anche etico-politiche, inducono ad esigere una giustificazione "pulita" di una conclusione giuridica<sup>6</sup>, che sia garantita la possibilità di un controllo preciso sul "ragionamento" di un sistema automatico.

Il particolare rilievo che spetta alla logica tra i linguaggi per la rappresentazione della conoscenza si riconnette anche al fatto che solo la logica offre i criteri cui far riferimento per valutare proposte alternative.

Nell'intelligenza artificiale sono stati proposti numerosi metodi per la rappresentazione della conoscenza. Ogni metodo ha trovato applicazione in settori determinati, fornendo spesso risultati soddisfacenti, ma è mancata finora una valutazione comparativa dell'adequatezza di ciascuna tecnica rispetto a compiti standardizzati. Alla presentazione delle notazioni adottate, e delle relative implementazioni spesso non corrisponde un attento esame del potere espressivo di ogni metodo proposto e della sua interpretabilità in diversi linguaggi.

Sono stati proposti numerosi linguaggi per la rappresentazione della conoscenza, che si prestano particolarmente all'uso dell'una o dell'altra tecnica. Anche per i linguaggi, manca un'analisi comparativa, una determinazione dei requisiti minimi che ogni linguaggio deve

---

<sup>4</sup>SOWA J.F., *Conceptual Structures*, p. cit., p. 25 osserva che "Forse Dio può costruire un agglomerato (kludge) che funziona, i poveri mortali hanno bisogno di una teoria che li guidi.

<sup>5</sup>E se ne conosce il comportamento nella misura in cui questo possa essere ridotto all'applicazione di una teoria (una base di conoscenza) determinata.

<sup>6</sup>Questo aspetto è stato trattato ampiamente nel cap. 6.

soddisfare, una discussione di problematiche fondamentali come quelle del rapporto di ciascun linguaggio con i concetti fondamentali della nostra tradizione logico-matematica, come quelli di verità, di teorema e di decidibilità. I pregi di ogni linguaggio spesso si ravvisano nelle complesse realizzazioni per le quali è stato usato, più che in caratteristiche valutabili in un comune quadro concettuale.

Alla moltiplicazione dei metodi e dei linguaggi per la rappresentazione della conoscenza corrisponde la proliferazione dei motori inferenziali: ogni sistema esperto (o ogni guscio) si presenta con il proprio motore inferenziale, che spesso, seppure progettato per un compito determinato, vanta una pretesa all'universalità: i pregi di ogni linguaggio spesso si ravvisano piuttosto nelle complesse realizzazioni per le quali è stato usato, che in caratteristiche valutabili in un comune quadro concettuale<sup>7</sup>.

La logica e il metodo deduttivo rappresentano la notazione per la rappresentazione della conoscenza e il metodo per la sua manipolazione, che meglio consentono di comparare e valutare notazioni e metodi alternativi<sup>8</sup>.

La logica presenta infatti importanti caratteristiche che nessun altro formalismo per la rappresentazione della conoscenza possiede, e, in particolare, un rigoroso impianto teorico nel quale impostare e controllare le problematiche della sintassi, della semantica, dei metodi di inferenza. D'altro lato alcuni dei problemi che hanno determinato l'adozione di linguaggi per la rappresentazione della conoscenza diversi dalla logica (problemi come il trattamento del ragionamento comune, della vaghezza ecc.) non hanno trovato in altri contesti soluzioni migliori di quelle ottenibili con metodi logici<sup>9</sup>.

---

<sup>7</sup>GÜNTHER F., LEHMANN H., SCHÖNFELD W., *A Theory for the Representation of Knowledge*, cit., pp. 40-41

<sup>8</sup>Si è autorevolmente osservato che "c'è un solo linguaggio adeguato alla rappresentazione della conoscenza (sia dichiarativa che procedurale), il linguaggio della logica dei predicati del primo ordine. C'è un solo modo intelligente di elaborare la conoscenza, cioè applicando metodi di inferenza deduttivi" (KOWALSKI R.A., in BRACHMAN R.J., SMITH B.C. (a cura di), *SIGART Newsletter: Special Issue on Knowledge Representation*, vol. 70, febbraio 1980).

<sup>9</sup>GÜNTHER F., LEHMANN H., SCHÖNFELD W., *A Theory for the Representation of Knowledge*, cit., p. 40 i quali osservano che ogni linguaggio di tipo logico porta con sé automaticamente (per il solo fatto di essere, appunto, un linguaggio logico) i seguenti vantaggi (che nessun altro schema per la rappresentazione della conoscenza possiede:

La logica offre un linguaggio adeguato all'uso delle principali tecniche per la rappresentazione della conoscenza adottate nell'intelligenza artificiale: come abbiamo visto, regole, reti semantiche e frames possano essere trascritte in linguaggi di tipo logico.

C'è da aggiungere che gli studi logici hanno condotto allo sviluppo di numerose logiche, che rappresentano estensioni o alternative della logica classica dei predicati del primo ordine (la "logica" per antonomasia). Così per risolvere problemi rispetto ai quali sia insufficiente la logica dei predicati del primo ordine si può pensare ad un'estensione della logica predicativa (ad esempio, una logica modale o deontica), o all'impiego di una logica di un ordine superiore. Questo aspetto è particolarmente importante per l'informatica giuridica che deve affrontare tutti i problemi del linguaggio comune. Pertanto, il linguaggio generale della logica è preferibile rispetto a linguaggi specializzati, più sintetici, iconici ed efficaci nel rappresentare tipi particolare di conoscenza<sup>10</sup>, ma incapaci di esprimere tutte le conoscenze rilevanti in contesti giuridici. La soluzione preferibile consiste, probabilmente, nel "definire linguaggi specializzati, come le tabelle, le reti semantiche e i frames nei termini del calcolo dei predicati. Avendo fatto ciò possiamo usare questi linguaggi dove sono appropriati; dove falliscono possiamo ritornare al potere espressivo dell'intero calcolo dei predicati"<sup>11</sup>. In questa prospettiva, i linguaggi specializzati non sono alternative rispetto alla logica, ma sintassi logiche alternative rispetto alla notazione standard.

---

-Sintassi: una nozione chiara di che cosa rappresenti una espressione ben formata, a tutti i livelli di complessità sintattica.

-Semantica: una chiara relazione con la struttura del mondo.

-Teorie di inferenza: una varietà di alternative equivalenti.

-Concetti di correttezza: non può essere provato nulla di falso.

-Capacità espressiva: è possibile determinare che cosa possa e che cosa non possa essere espresso in un sistema determinato.

-Distinzione tra logica proposizionale, logica del primo ordine, logiche di ordine superiore.

-Altri formalismi come il *lambda calculus*.

-Altre nozioni metalogiche relative alla decidibilità, la complessità di implementazione"

<sup>10</sup>Ad esempio, le reti semantiche per sono particolarmente adatte ad esprimere gerarchie concettuali, i frames per esprimere complessi rapporti tra oggetti e concetti.

<sup>11</sup>GENESERETH M.R., NILSSON N.J., *Logical Foundations of Artificial Intelligence*, Morgan Kaufmann, Los Altos (California), 1987, p. 40.

## 2. Due critiche all'uso di modelli logici nell'informatica giuridica

L'uso della programmazione logica nel diritto ha incontrato un crescente interesse. La relativa semplicità ed immediatezza con cui possono essere sviluppati programmi in PROLOG che trattano problemi giuridici e il rapporto tra logica e tradizione filosofico-giuridica hanno condotto alla realizzazione di numerose applicazioni informatico-giuridiche ispirate alla programmazione logica. Non sono mancate, tuttavia, alcune penetranti critiche contro l'impiego di modelli logici nelle applicazioni giuridiche dell'intelligenza artificiale.

### 2.1. La critica di Leith

Leith<sup>12</sup> muove dalla considerazione che nel diritto non esistono regole chiare (*clear rules*), ogni regola è suscettibile di diverse interpretazioni. Il processo di applicazione del diritto è un processo di negoziazione sociale, e non un processo basato su regole. Questo autore interpreta il processo di applicazione della legge come il risultato di un conflitto tra potere legislativo e potere giudiziario. Leith afferma inoltre che, in questo conflitto il vincitore è, di regola, il potere giudiziario<sup>13</sup>.

In questa prospettiva, l'uso di una rappresentazione logica del diritto si manifesta come il tentativo conservatore di irrigidire in diritto in una interpretazione determinata ed indiscutibile. Leith nega che "l'applicazione tecnica delle regole giuridiche" sia "il metodo migliore per raggiungere la giustizia"<sup>14</sup>, e che "la giustizia possa essere fondata mediante la logica". In questo modo "si libererebbe il giudice dalla responsabilità del giudizio". In conclusione, secondo Leith "i modelli logici sono inadeguati per trattare il diritto, poiché il problema vero

---

<sup>12</sup>LEITH P., *Clear Rules and Legal Expert Systems*, in MARTINO A.A., SOCCI NATALI F., *Automated Analysis of Legal Texts*, 1986, pp. 661-680; LEITH P., *Fundamental Errors in Legal Logic Programming*, in "The Computer Journal", 1986, pp. 545-552. Leith ha sviluppato il sistema ELI, già ricordato nel cap. 4 par. 3.5.

<sup>13</sup>Leith connette questa ricostruzione del processo di applicazione del diritto al rigetto delle posizioni filosofico-giuridiche di Hart e alla riproposizione dello stretto collegamento tra diritto e potere suggerita da Austin.

<sup>14</sup>LEITH P., *Fundamental Errors in Legal Logic Programming*, cit., p. 549.



del diritto non è la chiarificazione di termini individuali della legge, ma il controllo del potere giudiziario" e "i problemi dei programmatori logici sono il risultato di una falsa epistemologia: essi vedono il mondo nei termini di un modello computazionale e non riescono a stare al di fuori di quel modello. Pertanto ogni volta che cercano di applicare quel modello al mondo reale essi falliscono, poiché il mondo non è un mondo logico<sup>15</sup>.

## 2.2. La critica di Stamper

Una discussione più articolata dell'inadeguatezza del paradigma logicistico dell'intelligenza artificiale rispetto ai problemi giuridici è stata recentemente sviluppata da Ronald Stamper<sup>16</sup>.

Stamper osserva che "i sistemi esperti sono la creazione del paradigma dell'intelligenza artificiale che assume che una realtà oggettiva possa essere compresa e controllata dall'intelligenza di un individuo esperto che può essere sostituita da macchine"<sup>17</sup>.

Questo paradigma, adeguato, in linea di principio, nei contesti delle scienze naturali e matematiche, sarebbe invece pericoloso nelle scienze sociali, negli affari, e specialmente nel diritto. Per affrontare problemi sociali bisognerebbe muovere da diverse premesse, ammettere che: "la realtà è il prodotto soggettivo di essere umani che cercano di collaborare attraverso norme ed esperienze condivise, processo che può essere assistito ma non sostituito da elaboratori".

In questi contesti non vi sarebbe spazio per dei "sistemi esperti", se per sistemi esperti si intendano delle "scatole che contengono abbastanza conoscenza da costituire dei surrogati di esperti umani". Un sistema esperto deve essere solo un mezzo per delegare complesse decisioni di routine, basate su regole precise che debbano essere eseguite in modo burocratico da un tecnico o da un impiegato. Esso è "solo un aiuto meccanico alla comunicazione tra un esperto responsabile e un assistente

---

<sup>15</sup>Ivi, p. 552.

<sup>16</sup>Stamper è stato il promotore del progetto LEGOL, illustrato nel cap. 4 par. 2.5.

<sup>17</sup>STAMPER R.K., BACKHAUS J., ALTHAUSE K., *Expert Systems: Lawyers Beware*, relazione presentata al 4<sup>o</sup> Congresso internazionale sul tema "Informatica e regolamentazioni giuridiche", Roma, maggio 1988, p. 317. Questa problematica è ripresa e ampliata in STAMPER R.K., *The Role of Semantics in Legal Expert Systems and Legal Reasoning*, cit..

prudente"", quando è considerato come un surrogato per qualche persona remota (la fonte originale della "base di conoscenza") che delega la sua responsabilità alla macchina, allora il sistema esperto è un pericolo potenziale"<sup>18</sup>.

Il paradigma oggi prevalente nell'intelligenza artificiale, si baserebbe su un insieme di assunti metafisici nati nel mondo dell'informatica e inaccettabili in contesti giuridici e sociali, assunti che vengono assimilati attraverso una serie di metafore usate regolarmente e in modo aporetico:

### 2.2.1 La metafora del condotto

Le parole sono considerate come veicoli, condotti, per trasportare il significato da un posto all'altro. Si assume che i simboli siano portatori di una sostanza astratta ("la conoscenza", "l'informazione"). Non è necessario che qualcuno capisca la struttura del sistema "esperto": è sufficiente accumulare la conoscenza (i simboli) e il sistema penserà ad applicarla per risolvere i problemi. Si dimentica così il ruolo di una comprensione critica (nel dibattito della collettività degli specialisti) al fine del mantenimento di valida conoscenza, l'importanza del contesto sociale, la dipendenza dei significati dal processo di interpretazione, in ultima istanza da una cultura. Solo in tali contesti delle sequenze di caratteri diventano portatori di significato<sup>19</sup>.

L'uso della "metafora del condotto" fa dimenticare che, nel creare la base di conoscenza di un sistema informatico, non si compie una semplice parafrasi del testo originario, che ne conservi i significati, ma si passa ad un linguaggio formale<sup>20</sup>. L'uso (comune tra i programmatori PROLOG) di

---

<sup>18</sup>STAMPER R.K., BACKHAUS J., ALTHAUSE K., *Expert Systems: Lawyers Beware*, cit., pag. 322-333. Questi autori suggeriscono ai giuristi di essere sospettosi se la base della conoscenza nel suo insieme non può essere esaminata nel suo complesso ed essere il focus dell'attenzione critica di una comunità di specialisti (pag. 333).

<sup>19</sup>Ad esempio, per esprimere il rapporto tra padre e figlio possiamo usare il predicato padre di, ma sarebbe stato lo stesso usare invece della sequenza di caratteri padre di una qualsiasi altra sequenza di caratteri, ad esempio, padre, p, xfdgz ecc.

<sup>20</sup>Come si è osservato in precedenza, la base di conoscenza non è che una collezione di simboli sui quali il sistema può effettuare talune manipolazioni. Per il sistema informatico il significato di quei simboli non è quello attribuito loro attribuito nella nostra cultura, ma quello che risulta dalla sintassi del linguaggio e dal funzionamento del motore inferenziale.

espressioni tratte dal linguaggio naturale per formulare simboli trattati dal sistema come sequenze di caratteri arbitrari può essere ingannevole: si crea l'illusione che l'elaboratore capisca quelle espressioni<sup>21</sup>. Ogni ingegnere della conoscenza che contribuisca alla realizzazione del sistema può inventare simboli diversi per esprimere lo stesso concetto, simboli che saranno considerati dal sistema come diverse sequenze di caratteri: "la conoscenza non è più una struttura linguistica condivisa, ma una mistura di codici privati incommensurabili"<sup>22</sup>.

### 2.2.2 La metafora dell'elaborazione di dati

Nell'intelligenza artificiale si tende ad assimilare il funzionamento delle organizzazioni sociali a quello dei sistemi informatici: come in un sistema informatico vi è un flusso di elettroni tra i diversi componenti, così un sistema giuridico funziona attraverso un flusso di informazioni tra i diversi componenti che ricevono, elaborano e trasmettono informazione. In questa prospettiva, un sistema esperto è una sorta di colonna di distillazione che trasforma del materiale rozzo, i dati, in un materiale raffinato, l'informazione.

Il modello comunemente accettato è il seguente

**ingresso-----elaborazione-----uscita**

che viene trasposto ai sistemi sociali.

Si dimentica che la comunicazione sociale avviene invece secondo lo schema:

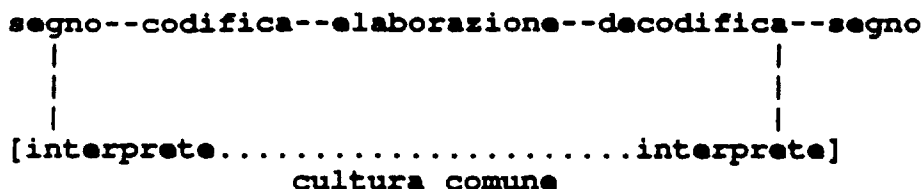
---

<sup>21</sup>Stamper et al. contrappongono la tecnica tradizionale del commento ai testi giuridici, della glossa, agli attuali sistemi esperti:

-il commento dell'esperto è volto a chiarire, a rendere più comprensibile il testo originario;

-il sistema esperto è volto invece a catturare e a fornire la competenza (expertise) come se fosse una merce. "E' una scatola chiusa che rende il campo della competenza più oscuro del necessario, impedendo o almeno rendendo più difficile la comprensione e la critica" (ivi, p. 326).

<sup>22</sup>Bisogna aggiungere che l'interpretazione dei documenti giuridici svolge una specifica funzione sociale, in relazione alla quale si differenzia dell'interpretazione di testi di altro tipo (cfr. 6. par. 4). La specifica competenza del giurista si rivela soprattutto nella soluzione di problemi interpretativi.



La comunicazione nei contesti sociali differisce profondamente da una trasmissione di segnali elettronici, in quanto si basa su processi di interpretazione basati su una cultura comune. Nel diritto i conflitti sociali si manifestano sotto forma di problemi di interpretazione.

### 2.2.3 La metafora insiemistica della realtà.

La semantica della logica dei predicati, e quindi quella dei linguaggi di programmazione logica (come il PROLOG) si basa su un modello di tipo insiemistico, e, in particolare, sui seguenti assunti:

- (1) Il mondo è diviso in individui discreti, e questa divisione è indipendente dall'osservatore.
- (2) Questi individui sono singolarmente identificabili.
- (3) Gli individui possono essere raggruppati in insiemi, la precisa appartenenza ai quali definisce il significato della proprietà alla quale corrispondono;
- (4) Coppie di individui possono essere raggruppati in insiemi allo stesso modo per definire le relazioni.

Invece, nel diritto, e, in generale, nella vita pratica, quando si considerino aspetti non di routine, i confini tra individui, le loro identità e le loro classificazioni sono aperti alla discussione<sup>23</sup>.

### 2.2.4 La metafora della corrispondenza per il significato e la verità

La logica matematica si basa su un concetto estensionale di significato: il significato di un termine è la sua estensione, l'insieme degli oggetti ai quali il termine si riferisce<sup>24</sup>.

<sup>23</sup>Circa l'identità, si pensi ai problemi dell'aborto e dei trapianti. Per quanto attiene ai problemi classificatori, si pensi alle continue discussioni sulla classificazione delle merci nel campo del diritto comunitario. Ad esempio, recentemente la Corte di giustizia delle Comunità europee ha deciso che, in contrasto con la legge italiana, la pasta secca di grano tenero deve ritenersi pasta alimentare

<sup>24</sup> Ad esempio, il significato di persona è l'insieme delle persone presenti (e eventualmente future) e il significato di un predicato come

Nella vita pratica questo concetto di significato è difficilmente utilizzabile (nessuno può pensare a tutte le persone o a tutti i rapporti di filiazione). Nel mondo della prassi può essere adottato solo un concetto di significato come intensione, come concetto correlato al termine. E' l'intensione del termine che fornisce il criterio per determinare se esso sia applicabile ad un certo individuo.

Ad una concezione del significato come estensione corrisponderebbe, secondo Stamper, una concezione della verità come corrispondenza: in base al significato estensionale delle espressioni che compaiono in una proposizione è possibile determinarne univocamente la validità e la falsità.<sup>25</sup>

Per il giurista invece "la realtà nella vita pratica rappresenta un consenso tra le parti interessate in un contesto (di cui è elemento chiave il sistema giudiziario) che assegna diversi poteri ad esse".

### 2.2.5 La metafora della mente come elaborazione di informazioni.

Gli studiosi di intelligenza artificiale sono portati a concepire la mente come un elaboratore, e quindi a concepire la percezione e il ragionamento come manipolazioni di simboli. Ciò implica l'assimilazione di ogni tipo di ragionamento alla manipolazione simbolica, alla deduzione logica. Nel ragionamento giuridico, invece, le inferenze deduttive hanno importanza limitata<sup>26</sup>.

## **3. Limiti dell'intelligenza artificiale giuridica**

Le critiche sviluppate da Leith e Stamper, risultano fondate, almeno in parte, se intese come critiche rivolte, in generale, verso i c.d. sistemi esperti giuridici. Infatti, è irrealistico pensare oggi a sistemi esperti giuridici in senso proprio, cioè a sistemi che siano in grado di fornire prestazioni che presuppongano notevoli competenze giuridiche, prestazioni analoghe a quelle che si richiedono ad un esperto giurista.

---

è **genitore di** è l'insieme delle coppie di persone delle quali l'una sia genitore dell'altra (<Gino, Carlo> <Amalia, Carlo> ecc.).

<sup>25</sup>Ad esempio, Gino è **genitore di** Carlo se e solo se la coppia <Gino, Carlo> appartiene all'insieme di coppie designato dal predicato **è genitore di**.

<sup>26</sup>Ivi, p.. 331.

Infatti le due attività che costituiscono il centro del lavoro giuridico - l'interpretazione delle norme e la qualificazione dei fatti rilevanti - non possono essere affidate ad alcun sistema informatico, non solo per ragioni etico politiche, ma anche per ragioni tecnologiche. Nessun sistema informatico, esistente o realizzabile allo stato della tecnica, sarebbe in grado di svolgere quelle funzioni<sup>27</sup>.

Tuttavia, l'identificazione tra legalismo e conservatorismo suggerita da Leith, ne l'acritica valorizzazione della creatività del giudice non è accettabile, per ragioni storiche, oltre che filosofico-giuridiche<sup>28</sup>. Il rapporto tra certezza del diritto e giustizia materiale rappresenta da sempre il problema fondamentale della filosofia e della pratica del diritto<sup>29</sup>. Qui importa sottolineare che l'uso di sistemi informatico-giuridici deve inserirsi nella dialettica tra norma e fatto, astratto e concreto, norma e interessi ecc., dialettica che caratterizza dell'esperienza etica e giuridica senza cancellare uno dei due termini di questa contrapposizione.

Come si è già osservato, l'uso di sistemi di conoscenza in grado di sviluppare inferenze logicamente corrette può riproporre questa dialettica, nei termini della contrapposizione tra deduzione dalla base di conoscenza, da un lato, e critica e eventuale modificazione della base di conoscenza che conduca a risultati "ingiusti", dall'altro.

I sistemi informatico-giuridici intelligenti vanno considerati strumenti per formulare con precisione, e quindi per applicare e controllare delle ipotesi di disciplina giuridica (una nuova legislazione o una determinata interpretazione del diritto in vigore). Si tratta di ipotesi da sottoporre ad un continuo, attento controllo alla luce delle loro conseguenze nei singoli casi.

---

<sup>27</sup>Questo tema è stato trattato ampiamente nel cap. 6.

<sup>28</sup>Come è noto, non è possibile collegare determinate in modo rigido posizioni politiche e approcci formalisti o antiformalisti nel diritto. Ad esempio, il fascismo italiano adottò una teoria giuridica ispirata al positivismo formalistico, mentre il nazionalsocialismo tedesco accolse invece un'impostazione antiformalistica; nei primi anni del regime comunista in Unione Sovietica prevalsero dottrine antiformalistiche, mentre in seguito si passò alla c.d. legalità socialista.

<sup>29</sup>Ci siamo brevemente soffermati su questo problema in precedenza. Qui mi limito ad osservare che non sembra corretto tradurre tale conflitto nel contrasto tra morale e diritto. Anche le esigenze della certezza e della prevedibilità del diritto si connettono a valori etici (cfr. SCARPELLI U., *Cosa è il positivismo giuridico*, cit.)

Il giurista, una volta individuate premesse che, in ipotesi, sembrano consentono di risolvere alcuni aspetti di una certa classe di problemi giuridici (ad esempio, la decisione della di responsabilità per incidenti automobilistici, il calcolo del canone dovuto in un rapporto di locazione, la determinazione della cittadinanza, il calcolo dell'ammontare di imposte ecc.) può rappresentare tali premesse nella base di conoscenza di un sistema informatico. Il giurista potrà poi concentrarsi sugli aspetti che è necessario o preferibile risolvere direttamente, caso per caso, e sui casi nei quali la risposta del sistema non è soddisfacente. Infatti, di fronte all'inadeguatezza materiale delle conseguenze della base di conoscenza (delle soluzioni proposte dal sistema per i singoli casi) il giurista potrà cercare soluzioni alternative. La generalizzazione di queste soluzioni si tradurrà nella proposta di una modifica della base di conoscenza, nei limiti in cui le nuove premesse (idonee a fondare quella conclusione) siano giuridicamente<sup>30</sup> e/o politicamente ammissibili<sup>31</sup>.

Ogni formalizzazione astratta di contenuti giuridici è suscettibile di condurre a risultati inadeguati, ingiusti, aberranti in taluni casi concreti<sup>32</sup>. Pertanto, l'uso di un sistema basato sulla conoscenza in contesti giuridici è ammissibile solo se la rigidità del meccanismo deduttivo trovi un rimedio nella possibilità valutare e modificare la base di conoscenza del sistema<sup>33</sup>:

(a) Deve essere garantita la trasparenza della base di conoscenza e del funzionamento del sistema, poiché la trasparenza è condizione necessaria della concreta possibilità di controllare e rivedere la base di conoscenza. La trasparenza presuppone:

---

<sup>30</sup>La disciplina giuridica dell'attività normativa e i rapporti gerarchici tra le diverse fonti del diritto limitano il campo delle proposte legislative, mentre i canoni dell'interpretazione giuridica limitano il campo delle interpretazioni ragionevolmente sostenibili

<sup>31</sup>La proposta di interpretazioni giuridiche avrà qualche possibilità di essere accolta solo nella misura in cui non si discosti eccessivamente dagli orientamenti di fatto prevalenti negli operatori del diritto (in particolare, nella magistratura).

<sup>32</sup>Cfr. cap. 2, par. 3.2.

<sup>33</sup>Qui intendo parlare solo dell'ammissibilità dell'uso di sistemi esperti giuridici in concreti contesti decisionali ed operativi, non della loro progettazione a fini di studio e ricerca. Inoltre, come già indicavo, i requisiti possono essere minori per sistemi destinati alla didattica.

-l'esplicitazione della relatività della ricostruzione delle norme e dei fatti dalla quale muove l'operazione inferenziale (chi ha operato la formalizzazione e a partire da quali fonti);.

-lo sviluppo di interfacce che agevolino la determinazione del contenuto della base di conoscenza, e quindi il suo riesame critico da parte di una cerchia di specialisti il più ampia possibile;

-l'esatta determinazione dei metodi di inferenza usati dal sistema;

-lo sviluppo di un meccanismo di spiegazione che consenta di ottenere una precisa giustificazione di ogni passo del processo inferenziale.

(b) Deve essere possibile, in relazione alla relatività e alla provvisorietà di ogni formalizzazione della conoscenza giuridica, un continuo aggiornamento della base di conoscenza. Ciò presuppone lo sviluppo di strumenti per intervenire sulla base conoscenza, realizzando le modifiche. Sarebbe importante, in particolare poter valutare l'impatto delle modifiche progettate sulla base di conoscenza (sulla sua consistenza, completezza, coerenza ecc.);

(c) Se un sistema esperto è uno strumento di comunicazione sociale, prima di introdurre un sistema esperto giuridico occorre predisporre un contesto organizzativo adeguato, che comprenda:

-la chiara definizione delle responsabilità di chi contribuisce alla preparazione della base di conoscenza del sistema;

-la definizione delle modalità d'uso del sistema: solo in relazione alla finalità determinata per la quale il sistema esperto è stato progettato. Il suo uso è giustificato solo nell'ambito di quella finalità.

#### **4. Limiti dell'approccio logico nel diritto**

Le critiche di Stamper e di Leith non sembrano invece giustificate in quanto critiche all'uso di un approccio logico nello sviluppi di sistemi basati sulla conoscenza giuridica. Ciò per due ordini di ragioni:

(a) In relazione all'universalità della logica e al fatto che essa rappresenta, come abbiamo visto, il modello cui sono riconducibili diversi formalismi dell'intelligenza artificiale, le obiezioni rivolte contro la logica, debbono essere intese come obiezioni rivolte contro la stessa possibilità (1) di una rappresentazione formale del diritto e (2) dell'uso di metodi deduttivi nel ragionamento giuridico e, quindi, (3) dello sviluppo di qualsiasi sistema informatico-giuridico che vada al di là del reperimento di testi.



(b) Proprio la logica, per la sua generalità e la sua solida base teorica, sembra lo strumento più adatto per lo sviluppo di sistemi informatici intelligenti che rispettino l'autonomia, l'iniziativa dell'uomo, che ne mettano in rilievo le capacità irriducibili al paradigma dell'intelligenza artificiale.

Un sistema informatico che si limiti ad inferenze logicamente corrette risulta uno strumento controllabile e limitato, i cui limiti mettono in rilievo la necessità di un'intervento umano.

Con lo sviluppo di un modello logico della normativa giuridica e/o dei fatti rilevanti per una certa classe di problemi giuridici ci si deve proporre di sviluppare un'ipotesi determinata, da valutare in tutte le sue implicazioni, non di sostituire alla complessità delle scelte etico-politica e della realtà sociale uno schema riduttivo. La definizione di un modello deduttivo può consentire una critica puntuale delle premesse dell'argomentazione giuridica<sup>34</sup>, è compatibile, come si è visto, anche con concezioni secondo le quali il ragionamento giuridico muove anche da premesse etiche<sup>35</sup>.

Le critiche riportate nel paragrafo precedente sono giustificate rispetto a sistemi che pretendono di comprendere i due aspetti seguenti (a) l'individuazione della premesse (normative e fattuali) della normativa giuridica e (b) la deduzione delle conseguenze di modelli logici di quelle premesse. Invece, sono ingiustificate rispetto ad un sistema che intenda limitarsi al secondo aspetto.

Oltre che allo sviluppo di un modello deduttivo determinato, il giurista può porsi l'obiettivo più ambizioso di formulare metaregole, regole sullo sviluppo di sistemi deduttivi. Si tratta di regole che conducono allo sviluppo di un'ipotesi determinata a partire da rappresentazioni che includano più scenari alternativi, più ricostruzioni alternative delle norme giuridiche e dei fatti (ed eventualmente, anche di regole che consentano di sviluppare alcune di dette rappresentazioni). In questo senso, i metodi per un trattamento informatico dell'incertezza (ragionamento analogico, probabilistico, per esempi, per difetto ecc.) senza intaccare la correttezza logica delle inferenze del sistema possono essere usati nel contesto della scelta tra diverse basi di conoscenza o della modifica e dell'adeguamento della base di conoscenza. Infatti, il principio della universalizzabilità del giudizio giuridico (ed etico) richiedono che di fronte ad un caso per il quale

---

<sup>34</sup>Costringendo ad esplicitare e precisare quelle premesse.

<sup>35</sup>Cfr. cap. 6 par. 3.2.

non vi sia una soluzione soddisfacente deducibile nel sistema, il giurista non si limiti ad adottare la soluzione proposta da un metodo impreciso. Egli deve procedere all'integrazione o alla modificazione delle premesse del sistema, cosicché ne discenda la soluzione di cui trattasi, in base a regole applicabili a tutti i casi dello stesso tipo<sup>36</sup>. Pertanto, i metodi automatici per il trattamento dell'incertezza nel diritto possono essere considerati come metodi per lo sviluppo della base di conoscenza piuttosto che come metodi che deviino dalla correttezza logica.

Per quanto attiene al concetto logico di verità e alla semantica estensionale della logica moderna, bisogna osservare che il problema sollevato da Stamper riguarda, più che la semantica logica, gli aspetti seguenti:

(a) l'esigenza di distinguere tra la verità e la sua ricerca, processo che conduce a risultati sempre parziali;

(b) l'esigenza di distinguere tra la verità di un enunciato e il fatto che quell'enunciato debba o possa, anche se possibilmente non vero (ad esempio, quando trovino applicazione regole di prova legale, o quando una sentenza abbia fatto stato) essere assunto quale premessa di determinate decisioni.

Invero:

(a) la concezione della verità come corrispondenza non impedisce, ma, in un certo senso, giustifica una concezione relativistica della conoscenza<sup>37</sup>.

(b) L'adozione di una semantica logica può consentire di ricostruire con il complesso rapporto tra verità di fatto e verità giuridica.

Più in generale, le distinzioni tra semantica intensionale e semantica estensionale, tra definizione aristotelica e definizione prototipica<sup>38</sup>, tra ragionamento analogico e ragionamento deduttivo, tra la definizione di determinate convenzioni e l'operare sulla base delle stesse, debbono essere considerate come polarità tra le quali si muove il pensiero umano e, in particolare, il pensiero giuridico.

---

<sup>36</sup>Regole sempre rivedibili di fronte alle peculiarità di nuovi casi.

<sup>37</sup>Come ha rilevato POPPER K.R., *On the Sources of Knowledge and Ignorance*, in *Proceedings of the British Academy*, 46, 1960, ripubblicato in *Conjectures and Refutations*, Routledge and Kegan Paul, London, 1963. Alcuni hanno invece osservato che è dubbio in quale misura il concetto tarskiano di verità coincida con il concetto di corrispondenza (cfr. HAACK S., *Filosofia delle logiche*, Franco Angeli, Milano, 1983, p. 126 ss.).

<sup>38</sup>Cfr. cap. 5 par. 2.3.

La logica (e le applicazioni informatiche della logica) possono rappresentare solo una di dette polarità. Pertanto, I sistemi basati sulla conoscenza, basati su modelli deduttivi, non rappresentano potenziali concorrenti del giurista, sono uno strumento utile ma limitato.

Inoltre, se è vero che i sistemi informatici intelligenti debbono essere tutti ricondotti allo stesso paradigma logicistico, la parzialità di questi sistemi non può trovare soluzione in metodi automatici, ma solo nell'interazione con l'uomo, e pertanto in un adeguato contesto organizzativo, che garantisca un rapporto equilibrato tra i due termini delle contrapposizioni che abbiamo illustrato. La considerazione delle prospettive e dei limiti dei sistemi informatico-giuridici intelligenti si conclude pertanto con un rinvio ad una problematica non affrontata nel presente volume, quella del contesto sociale e organizzativo dell'uso di tali sistemi.

Mi riprometto di affrontare in futuro questa complessa problematica. Il futuro dell'informatica giuridica si giocherà anche su questo piano, oltre su quello dei progressi scientifici e tecnologici.

1. Intelligenza artificiale e logica	268
2. Due critiche all'uso di modelli logici nell'informatica giuridica	273
2.1. La critica di Leith	273
2.2. La critica di Stamper	274
2.2.1 La metafora del condotto	275
2.2.2 La metafora dell'elaborazione di dati	276
2.2.3 La metafora insiemistica della realtà.	277
2.2.4 La metafora della corrispondenza per il significato e la verità	277
2.2.5 La metafora della mente come elaborazione di informazioni.	278
3. Limiti dell'intelligenza artificiale giuridica	278
4. Limiti dell'approccio logico nel diritto	281

## APPENDICE

### IL PROGRAMMA IRINORM

Nelle pagine seguenti si illustrerà IRINORM, un sistema che comprende strumenti per:

- (a) la rappresentazione della conoscenza giuridica nella logica dei predicati e la redazione di testi normalizzati;
- (b) l'uso della conoscenza formalizzata per effettuare analisi giuridiche.

IRINORM costituisce la prima realizzazione nell'ambito del già menzionato progetto IRI (Irnerio Regioni italiane). IRINORM rappresenta un primo risultato assai limitato, è servito più ad individuare taluni problemi che a trovar loro una soluzione. Tuttavia, esso presenta una struttura chiara, e si ispira ad un approccio logico. Esso consente di esemplificare alcuni dei concetti introdotti nelle pagine precedenti.

Il programma si inserisce in una più ampia ricerca sulle applicazioni dell'intelligenza artificiale al diritto (il progetto Irnerio), attivata presso il Centro interdipartimentale di ricerca in Filosofia del diritto e Informatica giuridica dell'Università di Bologna (CIRFID).

#### 1. La struttura del sistema

Il sistema IRINORM si divide in 2 sottosistemi:

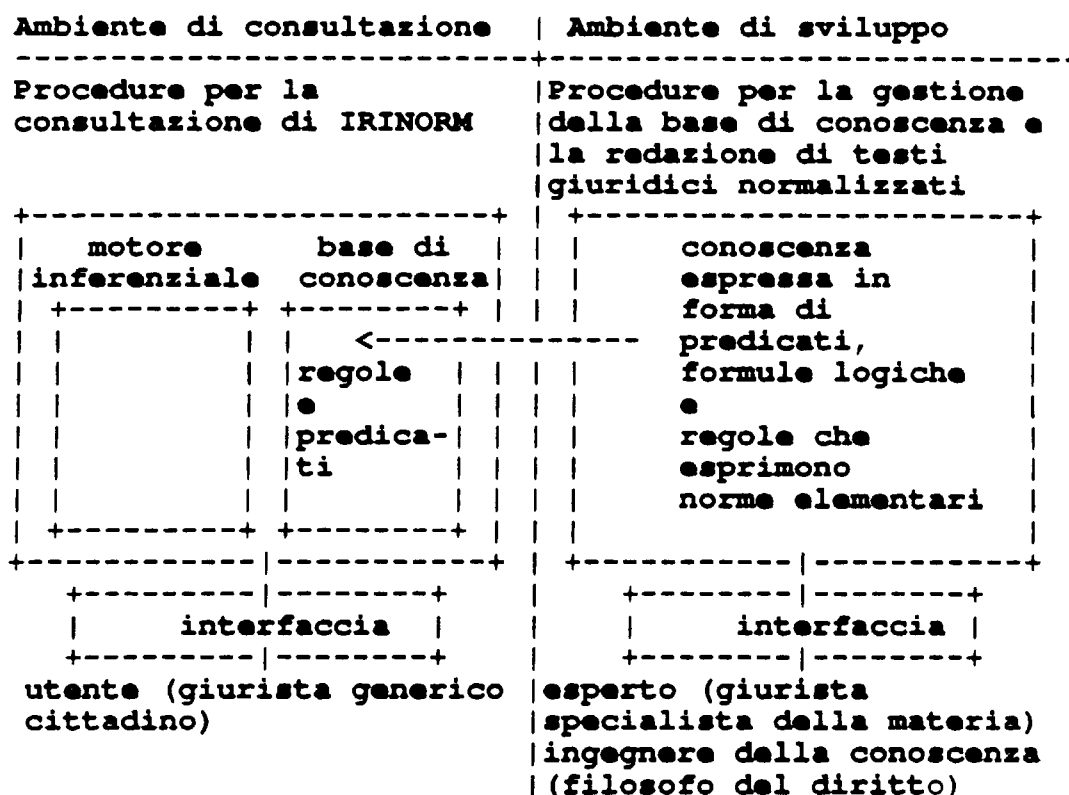
1) una procedura per l'analisi e la redazione di testi giuridici, e per la progettazione e la gestione della base di conoscenza;

2) una procedura per l'interrogazione del sistema, che utilizza la base di conoscenza e le informazioni sul caso fornite dall'utente per rispondere ai quesiti proposti.

Entrambe le procedure sono state realizzate in PROLOG, su microelaboratore. Esse rappresentano, seppure in forma embrionale, le due componenti fondamentali di un sistema basato sulla conoscenza:

- un ambiente di sviluppo (*development environment*) nel quale la conoscenza possa essere trasmessa al sistema, e
- un ambiente di consultazione (*consultation environment*) nel quale tale conoscenza possa essere utilizzata per risolvere dei problemi (cfr. fig. 1).

Fig. 1: schema del sistema IRINORM



## 2. Principi ispiratori

Nella realizzazione del sistema si sono seguite due fondamentali linee direttrici:

(a) l'uso della tecnica della normalizzazione, sviluppando le proposte di L.E. Allen, per l'analisi, la formalizzazione e la redazione di testi giuridici<sup>1</sup>;

(b) l'uso della logica predicativa come linguaggio per rappresentare contenuti giuridici.

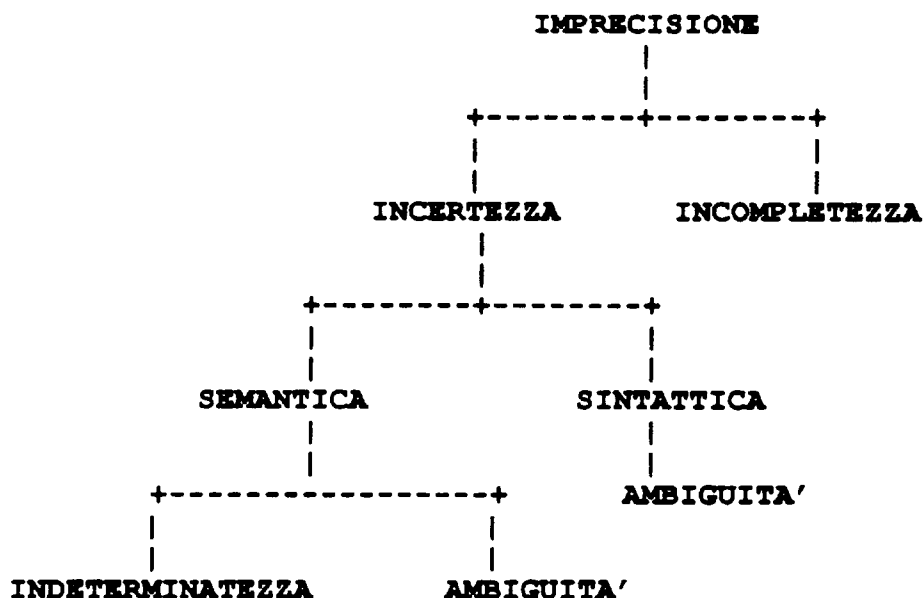
### 2.1 La tecnica della normalizzazione

Come è noto, la tecnica della normalizzazione fu ideata quale strumento per migliorare la stesura dei testi giuridici, eliminandone le ambiguità sintattiche.

<sup>1</sup>Il tema della normalizzazione è stato già toccato nel cap. 8 par. 1.1.

Seguendo le distinzioni proposte da Allen, e rappresentate nella fig. 2 si può dire che si ha una imprecisione in un enunciato o in un testo quando sia possibile - all'interno di determinati contesti linguistici e culturali - assegnare ad esso più di un significato<sup>2</sup>.

**Fig. 2. Tipi di imprecisione**



L'imprecisione può essere distinta in incertezza o incompletezza, a seconda che dipenda dalle espressioni effettivamente impiegate (che concorrono a formare l'enunciato) o dal fatto che sono state omesse talune espressioni<sup>3</sup>.

<sup>2</sup>LA trattazione della problematica dell'imprecisione del linguaggio giuridico avviene solitamente nell'ambito delle trattazioni dell'interpretazione, cioè nell'ambito di teorie intese a prescrivere come debbano essere risolti, o a descrivere come vengano di fatto risolti, i casi di imprecisione. Non è questo l'oggetto del presente lavoro. Qui ci si propone piuttosto di presentare strumenti che consentano di:

(a). esprimere con precisione le diverse possibili alternative interpretative, senza che abbia rilievo il fatto che l'una o l'altra debba essere o sia la prescelta,

(b). formulare un testo sintatticamente preciso, eventualmente al fine del suo uso quale premessa di inferenze automatiche.

<sup>3</sup>Il concetto di incompletezza non è molto chiaro, E' ovvio che ogni enunciato impreciso potrebbe essere precisato ampliandolo con ulteriori

L'incertezza può essere ulteriormente distinta in incertezza sintattica e incertezza semantica, a seconda che concerna gli aspetti sintattici o gli aspetti semantici dell'enunciato.

Per aspetti sintattici dell'enunciato, nell'accezione qui usata, si intendono quelli che attengono al modo nel quale i vocaboli sono combinati nella struttura dell'enunciato. Tra gli aspetti sintattici comprendo inoltre quelli relativi al significato delle parole che esprimono rapporti sintattici. Pertanto, sono incertezze sintattiche, le incertezze strutturali in senso proprio (cioè quelle relative alla combinazione dei vocaboli), e inoltre tra le incertezze lessicali (relative al significato dei singoli vocaboli), quelle che attengono al significato delle espressioni sintattiche. L'incertezza sintattica assume, di regola, la forma dell'ambiguità sintattica<sup>4</sup>: le caratteristiche grammaticali dei vocaboli, l'ordine nel quale compaiono e i significati attribuibili ai termini sintattici sono insufficienti a determinare univocamente gli aspetti sintattici di un enunciato, di modo che a questo risultano ascrivibili più strutture sintattiche alternative<sup>5</sup>.

---

espressioni e in questo senso, ogni enunciato impreciso può dirsi incompleto. Forse potremmo intendere l'incompletezza come un'imprecisione di tipo "contestuale": l'enunciato può ricevere, alla luce del contesto nel quale è inserito, un significato diverso da quello che gli sarebbe attribuibile senza considerare quel contesto, e manca nell'enunciato stesso un riferimento a questo influsso del contesto nel suo significato (in questo senso l'enunciato è incompleto). E' questo il caso, ad esempio, delle eccezioni senza riferimento incrociato.

<sup>4</sup>A mio parere, le incertezze sintattiche di tipo lessicale possono assumere anche la forma dell'indeterminatezza. Quando un vocabolo sintattico non viene interpretato come un connettivo logico (ad esempio, la congiunzione "poiché" usata per esprimere un rapporto di causalità), può essere difficile determinare in modo indiscutibile se esso sia applicabile alla relazione tra certi fatti (si pensi al problema della causalità nell'illecito). Accanto a problemi di ambiguità (ad esempio, la scelta tra diversi concetti di causalità), possono verificarsi ipotesi di indeterminatezza (pur usando un determinato concetto di causalità - ad esempio, la cosiddetta causalità umana - non è chiaro se esso si applichi o meno al rapporto tra due fatti determinati - ad esempio, al rapporto tra un determinato incidente automobilistico ed una malattia della vittima).

<sup>5</sup>Il concetto di struttura sintattica del linguaggio, così come quello la distinzione tra sintassi e semantica, e il rapporto tra struttura sintattica e struttura logica sollevano problemi troppo ampi e complessi per poter essere affrontati in questa sede. Mi limito ad osservare che muovo dalla premessa che ogni enunciato grammaticalmente corretto



Per aspetti semantici di un enunciato, nell'accezione qui usata, si intendono invece quelli che attengono ai significati dei vocaboli non sintattici che compaiono nell'enunciato, cioè dei vocaboli che designano gli oggetti del discorso e le loro proprietà e relazioni. L'incertezza semantica può assumere le due forme della indeterminatezza (o vaghezza) e dell'ambiguità. C'è un'indeterminatezza semantica quando non si sia in grado di determinare con esattezza l'estensione di un termine, cioè di determinare, in relazione ad un significato di quel termine, il confine tra ciò cui esso si riferisce e ciò cui esso non si riferisce. C'è invece un'ambiguità semantica quando un termine abbia più significati nettamente distinguibili.

Come è noto, il linguaggio giuridico (così del legislatore come della dogmatica giuridica) non è un linguaggio formalizzato. Infatti, tale linguaggio prende a prestito dal linguaggio comune il lessico, la sintassi e i significati dei termini usati.

Il linguaggio giuridico, come tutti i linguaggi specialistici, conosce la possibilità di una tecnicizzazione, cioè dell'impiego di vocaboli (o espressioni) tecnici e di vocaboli (o espressioni) tecnicizzati. Tuttavia, tale tecnicizzazione è solitamente limitata ai termini semantici, e non si estende ai termini che esprimono rapporti sintattici. Per esprimere le strutture sintattiche, si fa invece completo affidamento sul linguaggio comune. In particolare, nel linguaggio giuridico non si verifica neppure quel fenomeno di una tecnicizzazione delle espressioni sintattiche che caratterizza il linguaggio scientifico, dove il frequente impiego di espressioni del linguaggio naturale e di

---

(enunciato grammaticale) sia dotato di una struttura sintattica, di una determinata combinazione dei componenti (frasi, parole, ad un livello più profondo, formativi, cioè unità ultime sintatticamente rilevanti), che funge da tramite tra la sequenza di suoni o di segni grafici che esprimono l'enunciato e il suo significato. Le strutture sintattiche sono state al centro degli studi linguistici negli ultimi decenni. Bisogna ricordare, in particolare, la c.d. grammatica generativa (cfr. CHOMSKI N., *Aspects of the Theory of Syntax*, M.I.T. Press, Cambridge (Mass), 1965), teoria linguistica che, come è noto, ha consentito di collegare lo studio dei linguaggi formali e dei linguaggi naturali e ha costituito la base di partenza della linguistica computazionale, e le grammatiche di Montague (MONTAGUE C., *Formal Philosophy*, Yale University Press, New Haven, 1974) che rappresentano il più importante contributo per una traduzione del linguaggio naturale in una struttura logiche. In seguito, importanti studi sulle strutture linguistiche, non ancora recepiti nella nostra cultura giuridica, sono stati compiuti (anche) in funzione di applicazioni informatiche, nell'ambito delle problematiche della rappresentazione della conoscenza e del trattamento del linguaggio naturale in sistemi automatici.

espressioni logico-matematiche nei medesimi contesti, determina una tendenza ad interpretare le prime come sinonimi delle seconde.

Il linguaggio giuridico è quindi caratterizzato da una notevole imprecisione sintattica. Questa imprecisione raramente produce gravi problemi interpretativi, ma spesso appesantisce la comprensione del testo, e impedisce al giurista di concentrarsi sui problemi interpretativi sostanziali (che attengono solitamente all'indeterminatezza dei termini semantici). L'indeterminatezza<sup>6</sup> delle espressioni impiegate dal legislatore può rispondere ad importanti esigenze giuridiche e sociali (evitare un eccessivo spezzettamento casistico della disciplina, favorire un adeguamento giurisprudenziale alle particolarità del caso e al mutare delle condizioni sociali ...), e può essere quindi consapevolmente introdotta dal legislatore stesso. Ciò non vale, di regola, per le ambiguità e, in particolare, sembra difficile pensare alla preordinata introduzione di un'ambiguità sintattica in un testo legislativo<sup>7</sup>.

---

<sup>6</sup>Che è, in qualche misura, conseguenza inevitabile della generalità), come si è visto nel cap. 5 par. 2.4.

<sup>7</sup>Il tipo più frequente di ambiguità sintattica nei testi legislativi attiene probabilmente alla natura tassativa o meno dell'elencazione delle condizioni alle quali è subordinato una certa qualità o situazione giuridica. Si pensi, ad esempio all'art. 1429 del codice civile, che esprimiamo come segue, riportando una sola volta il connettivo principale "quando":

"L'errore è essenziale quando:

- (1) cade sulla natura o sull'oggetto del contratto;
- (2) cade sull'identità dell'oggetto della prestazione o su una qualità dello stesso che, secondo il comune apprezzamento o in relazione alle circostanze, deve ritenersi determinante del consenso;
- (3) cade sull'identità o sulle qualità della persona dell'altro contraente, sempre che l'una o le altre siano state determinanti del consenso;
- (4) trattandosi di errore di diritto, è stato la ragione unica o principale del contratto.

Un primo problema di interpretazione sintattica, di ovvia soluzione e quindi puramente teorico, attiene al significato dei punti e virgola che separano le diverse caratteristiche dell'errore essenziale elencate dall'articolo. Se leggiamo quei punti e virgola come delle congiunzioni, degli "e" logici (in simbolo "&"), allora un errore dovrà ritenersi essenziale solo quando ricorrano tutte le condizioni di cui ai numeri da 1 a 4, cioè solo nell'ipotesi assai remota che l'errore cada sulla natura o sull'oggetto del contratto, e contemporaneamente anche sull'identità dell'oggetto della prestazione o su una qualità dello stesso determinante del consenso, e contemporaneamente anche

Per risolvere queste ambiguità, Allen propone di ridefinire le espressioni sintattiche e, specificamente, di introdurre nel linguaggio giuridico i connettivi logici. L'uso di espressioni sintattiche definite logicamente si unisce ad un'ipotesi circa la struttura del linguaggio normativo: ogni norma è rappresentabile con un'implicazione (un enunciato ipotetico) e pertanto ogni enunciato normativo deve essere equivalente ad una congiunzione di implicazioni.

Si è sopra osservato come l'ambiguità sintattica possa dipendere non dall'incertezza del significato delle espressioni sintattiche ma anche dall'incertezza circa la struttura dell'enunciato. In particolare, l'ambiguità strutturale in senso proprio può riguardare la determinazione degli argomenti delle espressioni sintattiche, il "campo" cui queste si riferiscono. Per eliminare questo secondo tipo di ambiguità Allen propone alcune tecniche grafiche che consentono di specificare univocamente ciò cui si riferisce ogni espressione sintattica formalizzata, e che svolgono quindi una funzione analoga a quella delle parentesi nelle espressioni logiche o matematiche.

## 2.2 L'uso della logica per esprimere contenuti giuridici

La tecnica della normalizzazione conduce alla redazione di un testo in cui parte della struttura sintattica è espressa nel linguaggio della logica.

Ciò comporta modifiche rispetto al corrispondente testo in linguaggio naturale, ma non così radicali come si potrebbe pensare. In particolare, la normalizzazione di un testo giuridico non comporta l'assegnazione di un preciso

---

sull'identità o le qualità dell'altro contraente ecc. Se invece (come suggeriscono la ripetizione del verbo e l'enumerazione delle diverse ipotesi) leggiamo i punti e virgola come delle disgiunzioni, degli "o" logici (in simbolo " $\vee$ ") dovrà considerarsi sufficiente il verificarsi di quanto contemplato da uno dei punti dell'enumerazione (ad esempio, che l'errore cada sull'identità dell'oggetto del contratto) affinché l'errore debba ritenersi essenziale.

E' più delicato un secondo problema di interpretazione sintattica, che attiene al significato del connettivo "quando". Se leggiamo quel connettivo attribuendogli il significato di una semplice implicazione, un "se" logico (in simbolo " $\rightarrow$ "), allora il realizzarsi di una delle ipotesi contemplate dall'articolo è condizione sufficiente affinché l'errore sia essenziale, ma non necessariamente le ipotesi di errore essenziale si riducono a quelle previste dall'art. 1429 cc. Se invece leggiamo il connettivo di cui trattasi come una doppia implicazione, un "se e solo se" logico (in simbolo " $\leftrightarrow$ "), allora le ipotesi di errore essenziale di cui all'art. 1429 esauriranno i casi di errore essenziale (la previsione dell'articolo è tassativa).

significato ad ogni vocabolo. Infatti, la formalizzazione è limitata alle strutture sintattiche prese in considerazione, e tocca solo marginalmente il problema dell'ambiguità e dell'indeterminatezza semantica.

Quando alla formalizzazione dei termini sintattici si unisca l'uniformità nell'uso delle espressioni non formalizzate, si realizzano le condizioni sufficienti per l'applicazione di un calcolo logico. Per quanto riguarda le espressioni non formalizzate, e, in particolare, le espressioni semantiche, si richiede solo che esse siano impiegate sempre nel medesimo significato nell'ambito del testo normalizzato<sup>8</sup>.

Infatti, l'impiego dei metodi della logica per valutare la correttezza di un'inferenza o per sviluppare (anche automaticamente) un'inferenza a partire da premesse date, presuppone che sia espressa nel linguaggio della logica solamente la struttura logica degli enunciati che compaiono nell'inferenza da valutare o nelle premesse da cui muovere. Il concetto di struttura logica è relativo al livello cui si spinge l'analisi logica: le espressioni da tradurre nel linguaggio formale aumentano quanto più si voglia approfondire l'analisi dell'inferenza o accrescere la capacità di effettuare inferenze applicando le leggi della logica.

Così per l'applicazione del calcolo proposizionale è sufficiente la formalizzazione delle espressioni (come "e", "non", "o", "se", "a meno che non", "ma altrimenti" ecc.) che riteniamo di poter interpretare come connettivi vero-funzionali tra enunciati, e delle modalità con cui si identificano gli argomenti di quei connettivi.

Ai fini dell'applicazione del calcolo dei predicati, la formalizzazione deve toccare anche le espressioni (come "tutti", "alcuni", "qualche" ecc.) traducibili in quantificatori e le modalità con le quali si esprime il rapporto tra i predicati e i termini ai quali si riferiscono.

Per l'applicazione della logica modale, la formalizzazione deve estendersi anche alle espressioni ("è

---

<sup>8</sup>Sull'esigenza di una siffatta uniformità nell'uso linguistico, cfr., ALEXI R., *Theorie der juristischen Argumentation*, cit., 1978, p. 234 ss.. Non esigiamo che i termini del linguaggio usato siano privi di indeterminatezza, cioè che si sia sempre in grado di determinare con sicurezza se un'espressione si applichi o no ad un certo oggetto. Ci limitiamo ad esigere che, qualora una certa occorrenza di un'espressione non formalizzata (quindi di un enunciato atomico al livello della logica proposizionale, e di un predicato, funzione o costante al livello della logica dei predicati) si applichi ad un certo fatto od oggetto (serie di oggetti), si sia disposti ad ammettere che anche ogni altra occorrenza di quell'espressione si applichi a quel fatto od oggetto.

possibile che", "è necessario che" ecc.) traducibili mediante gli operatori modali, per l'applicazione della logica deontica, anche alle espressioni ("è obbligatorio che", "è permesso che" ecc.) traducibili mediante operatori deontici, e così via.

La formalizzazione conduce ad un duplice risultato:

-la redazione di un testo normalizzato, in un linguaggio abbastanza vicino al linguaggio naturale, ma logicamente rigoroso,

-la preparazione della base della conoscenza del sistema.

L'impiego di espressioni logicamente definite comporta infatti due importanti conseguenze:

-l'eliminazione di alcune imprecisioni sintattiche del testo originario (di quelle che discendevano dall'ambiguità lessicale dei vocaboli sostituiti con le espressioni logiche o dall'ambiguità strutturale relativa alla determinazione degli argomenti di quei vocaboli);

-la possibilità di utilizzare strumenti informatici per valutare o per effettuare delle inferenze.

### **3. L'ambiente di sviluppo**

Nell'ambiente di sviluppo la conoscenza va formalizzata nel linguaggio della logica dei predicati. A partire da questa rappresentazione è possibile redigere automaticamente testi normalizzati in diverse lingue e in più versioni logicamente equivalenti.

Pertanto, nelle pagine seguenti si illustreranno le seguenti funzioni:

1. l'inserimento e la modifica della conoscenza;
2. la redazione di testi normalizzati;
3. la generazione della base di conoscenza.

#### **3.1 L'inserimento e la modifica della conoscenza**

I contenuti giuridici vengono rappresentati mediante due componenti:

a. i predicati (ciascuno dei quali è contraddistinto da un breve identificatore);

b. le formule logiche (anch'esse contrassegnate da un identificatore), che sono formule della logica dei predicati. Più esattamente si usa un sottoinsieme della logica dei predicati che forma un'estensione delle note

clausole di Horn (ci si propone di sviluppare il sistema così da poter usare l'intera logica dei predicati)<sup>9</sup>.

I predicati vengono memorizzati nella seguente sintassi<sup>10</sup>

**"<identificatore del predicato>","{(<testo del  
predicato>) {(<numero di argomento>)>}"**

dove identificatore del predicato è l'identificatore del predicato, testo del predicato è una parte dell'espressione che esprime il predicato nel linguaggio naturale, numero di argomento è un'espressione numerica: sostituendo termini del linguaggio formale (per ora solo costanti e variabili) a queste espressioni si ottiene un enunciato atomico. E' possibile inserire i numeri di argomento all'interno del predicato, così da esprimere (come fa il linguaggio naturale) la funzione di ciascun argomento nella struttura sintattica dell'enunciato. Ad esempio, al seguente predicato:

**"r7","(1) risarcisce (2) per il danno (3)"**

corrispondono enunciati come:

**r7","(antonio) risarcisce (marco) per il danno  
(distruzione archivi)"**

o come

**"r7","(P1) risarcire (P2) per il danno (D)".**

Le cifre indicano quindi le posizioni nelle quali si dovranno inserire dei termini (variabili o costanti) per ottenere un enunciato. Nella fig. 3 riportiamo alcuni predicati in materia di responsabilità civile.

---

<sup>9</sup>Sulle clausole di Horn, cfr. cap. 7 par. 2.

<sup>10</sup>Per descrivere la sintassi nella la quale formalizziamo la conoscenza giuridica, adottiamo il formalismo di Bakus Narus, frequentemente usato per descrivere i linguaggi di programmazione. In tale formalismo, i termini sintattici vengono denotati da espressioni racchiuse tra parentesi angolari (ad esempio, <termine>), le parentesi graffe indicano 0 o più ripetizioni dei simboli racchiusi, A|B indica l'alternativa tra l'espressione A e l'espressione B. Ad esempio

**[<vuoto>|<termine> {,<termine>}]**

indica che le parentesi quadre racchiudono un'espressione vuota, oppure un termine, seguito eventualmente da altri termini, ciascuno separato con una virgola dal termine precedente (come [] o [A] o [A,B,D] ecc.).

**Fig. 3: predicati**

- ("r1", "un fatto (1) ha causato a (2) un danno (3)")
- ("r2", "(1) ha compiuto (2)")
- ("r3", "(1) è doloso")
- ("r4", "(1) è colposo")
- ("r5", "il fatto (1) da parte di (2) è ingiusto verso (3)")
- ("r6", "concorre con il fatto (1) una causa di giustificazione")
- ("r7", "(1) risarcisce (2) per il danno (3) in seguito al fatto (4)")
- ("r8", "(1) è arrecato nell'esercizio di un diritto di (2)")
- ("r9", "(1) è lesione di interesse tutelato giuridicamente di (2)")
- ("r10", "(1) è lesione di un diritto di (2)")
- ("r11", "concorre con (1) legittima difesa")
- ("r12", "concorre con (1) stato di necessita")
- ("r13", "concorre con (1) forza maggiore")

Le formule logiche si esprimono nella seguente sintassi:

"<nome>", "<formula>",

dove una formula può essere:

- <enunciato elementare>,
- n <formula>
- <formula><connettivo><formula>,
- <operatore deontico> <formula>.

Il simbolo **n** rappresenta la negazione, **connettivo** è uno dei connettivi binari riportati alla fig. 1 del cap. 8, **operatore deontico**, per ora, è solamente "o" (è obbligatorio che) e "p" (è permesso che). Ciascun connettivo può essere espresso mediante un simbolo, da impiegarsi nelle formule logiche, o mediante un'espressione più vicina al linguaggio naturale (in quest'ultimo caso lo scriviamo in maiuscolo per distinguerlo dall'altrimenti omonima espressione in linguaggio naturale), da impiegarsi nel testo normalizzato<sup>11</sup>.

Nelle formule logiche, gli enunciati elementari interpretativi si esprimono nella seguente sintassi:

<identificatore del predicato> [<vuoto> | <termine>,  
<termine>],

<sup>11</sup>I connettivi di Irinorm sono stati riportati nel cap. par.^.

dove **termine** può essere, per ora, solo una variabile o una costante. Gli enunciati nei quale compaiono variabili si interpretano assumendo che tutte le variabili siano quantificate universalmente. Infatti, non è ancora stato sviluppato un trattamento della quantificazione esistenziale e dei termini complessi (funzioni).

E' possibile utilizzare il sistema anche per gestire formule della logica proposizionale. Gli enunciati atomici di tali formule vengono semplicemente interpretati come predicati a 0 argomenti.

Nella fig. 4 riportiamo alcune formule logiche in materia di responsabilità civile, nelle quali compaiono le etichette dei predicati della fig. 3. Le formule sono interpretate assumendo che tutte le variabili siano quantificate universalmente (non abbiamo ancora sviluppato un trattamento della quantificazione esistenziale).

**Fig. 4: formule logiche**

```
f("cc. 2043", "((r1[F,P2,D] & r2[P1,F] & (r3[F] v r4[F])
& r5 [F,P1,P2] ) > r7[P1,P2,D,F]) sc r6[F]")
f("ingiustizia", "r5[D,P1,P2] se (n r8[D,P1] & (r9[D,P2]
v r10[[D,P2]))")
f("giustificazione", "r6[F] eq (r11[F] v r12[F] v
r13[F])")
```

### 3.1.1 La redazione di testi normalizzati

IRINORM consente di redigere automaticamente testi normalizzati, a partire dai predicati e dalle formule interpretative introdotte nel sistema.

Il testo normalizzato è composto di enunciati atomici, connettivi logici e operatori deontici combinati nel modo specificato dalle corrispondenti formule logiche. Gli enunciati atomici sono costruiti sostituendo i numeri di argomento nel predicato con i termini (variabili e costanti) che compaiono nella formula logica (l'n-esimo termine sostituisce il numero n). Ad esempio, la prima formula della fig. 4, può essere tradotta nel testo normalizzato della fig. 5. Ciascun predicato (atomico o molecolare) è contrassegnato da una lettera o da un numero. Ogni connettivo ha per argomenti gli enunciati contrassegnati dalle lettere o dai numeri allineati (indentati) con il connettivo stesso.



**Fig. 5: testo normalizzato**

**1. SE**

- A. 1. [r1 : un fatto {F} ha causato a {P2} un danno {D}], E
- 2. [r2 : {P1} ha compiuto {F}], E
- 3. A. [r3 : {F} è doloso], O
- B. [r4 : {F} è colposo], E
- 4. [r5 : {F} da parte di {P1} è ingiusto verso {P2}]

**ALLORA**

**B. E' OBBLIGATORIO CHE**

- [r7 : {P1} risarcisce {P2} per il danno {D} in seguito al fatto {F}]

**SALVO CHE**

- 2. [r6 : concorre con {F} causa di giustificazione]

In futuro, seguendo le indicazioni di ALLEN<sup>12</sup>, ci si ripromette di consentire la scelta tra diverse redazioni del testo normalizzato (con o senza etichette, con o senza indentazione, ecc.).

Già oggi è possibile ottenere automaticamente testi normalizzati in più lingue diverse. Per ciascun predicato si dovrà introdurre una traduzione nelle lingue cui si è interessati (usando la medesima etichetta). Il sistema provvede quindi alla redazione del testo normalizzato nella lingua richiesta. Questa funzione di IRINORM potrebbe risultare utile per predisporre o applicare in modo uniforme testi giuridici redatti in lingue diverse (si pensi, ad esempio, alle Comunità Europee, dove i testi giuridici debbono essere redatti in 9 lingue ufficiali). Nella fig. 6 possiamo vedere la traduzione inglese del testo normalizzato riportato alla fig. 5.

---

<sup>12</sup>Cfr. ALLEN L.E., SAXON, *Computer Aided Normalizing and Unpacking: some Interesting Machine-processable Transformations of Legal Rules*, cit.

**Fig. 6: testo normalizzato in traduzione inglese**

```

1. IF
  A. 1. [r1 : a fact {F} caused damage {D} to {P2} ], E
     2. [r2 : {P1} did {F}], E
     3. A. [r3 : {F} is intentional], O
        B. [r4 : {F} is negligent], E
     4. [r5 : {F} by {P1} is unlawful as regards {P2}]
  THEN
  B. IT IS OBLIGATORY THAT
     [r7 : {P1} compensates {P2} for the damage {D}]
UNLESS
2. [r6 : a defence is concurrent with {F}]

```

Al fine di mettere in rilievo la struttura logica del testo normalizzato, può essere redatto automaticamente uno schema del testo normalizzato nel quale gli enunciati elementari sono sostituiti dalle corrispondenti etichette.

**Fig. 7 : schema del testo normalizzato**

```

1. IF
  A. 1. r1[F,P2,D]) E
     2. r2[P1,F]) E
     3. A. r3[F]) O
        B. r4[F]) E
     4. r5[F,P1,P2])
  THEN
  B. r7[P1,P2,D]
UNLESS
2. r6[F]

```

Possono essere redatte automaticamente diverse versioni logicamente equivalenti del medesimo testo normalizzato.

Seguendo le indicazioni di ALLEN chiamiamo:

-Elementare la struttura logica nella quale compaiano solo i connettivi "SE-ALLORA", "NON SI VERIFICA CHE", "E";

-Fondamentale la struttura logica nella quale oltre a connettivi presenti nella versione fondamentale compaiano solo i connettivi "O" e "AUT";

-Chiara la struttura logica nella quale, oltre a connettivi presenti nella struttura fondamentale, compaiano solo i connettivi "SE", "SE E SOLO SE", "ANCHE SE", "MA ALTRIMENTI", "MA ALTRIMENTI NO";

-Complessa la struttura logica nella quale oltre a connettivi presenti nella struttura chiara compaiano anche i connettivi "SALVO CHE", "SALVO CHE1" e "SALVO CHE2".

Si può ottenere una disposizione normalizzata con la struttura logica prescelta, logicamente equivalente alla

disposizione normalizzata corrispondente alla formula logica immessa dall'utente.

La versione normalizzata con struttura logica elementare consiste di una congiunzione di enunciati elementari, di negazioni di enunciati elementari e di implicazioni, ciascuna delle quali ha per antecedente una congiunzione di enunciati elementari o di negazioni di enunciati elementari e per conseguente un enunciato elementare o la negazione di un enunciato elementare (nella fig. 7 riportiamo la versione elementare del testo normalizzato di fig. 5)

**fig. 7: testo normalizzato con struttura logica semplificata**

1. SE
  - A. 1. NON SI VERIFICA CHE
    - A. [r6 : concorre con {F} causa di giustificazione], E
    2. [r1 : un fatto {F} ha causato a {P2} un danno {D}], E
    3. [r2 : {P1} ha compiuto {F}], E
    4. [r3 : {F} e doloso], E
    5. [r5 : {F} da parte di {P1} e ' ingiusto verso {P2}]
  - ALLORA
  - B. E' OBBLIGATORIO CHE
    - [r7 : {P1} risarcisce {P2} per il danno {D} in seguito al fatto {F}], E
2. SE
  - A. 1. NON SI VERIFICA CHE
    - A. [r6 : concorre con {F} causa di giustificazione], E
    2. [r1 : un fatto {F} ha causato a {P2} un danno {D}], E
    3. [r2 : {P1} ha compiuto {F}], E
    4. [r4 : {F} e colposo], E
    5. [r5 : {F} da parte di {P1} e ' ingiusto verso {P2}]
  - ALLORA
  - B. E' OBBLIGATORIO CHE
    - [r7 : {P1} risarcisce {P2} per il danno {D} in seguito al fatto {F}]

Ciascuno degli enunciati membri della congiunzione che costituisce la versione elementare esprime una norma elementare. Assumiamo infatti, seguendo un'opinione diffusa

nella cultura filosofico-giuridica, che tutte le norme siano riducibili ad una congiunzione di norme di questo tipo<sup>13</sup>.

La trasformazione di un qualsiasi arbitrario enunciato in una struttura sintattica di questo tipo può condurre a risultati controintuitivi. Il problema è particolarmente grave in contesti giuridici, in quanto l'implicazione normativa sembra differire dall'ordinaria implicazione materiale, e avvicinarsi alla cosiddetta implicazione stretta della logica modale o ad analoghi formalismi idonei ad esprimere rapporti causali. Si è data una risposta provvisoria al problema individuando un insieme di trasformazioni tali da condurre a formule logicamente equivalenti, ma nelle quali siano conservati i rapporti di condizionamento espressi nelle formule di partenza<sup>14</sup>.

Possono essere ottenuti automaticamente schemi dei testi normalizzati con struttura logica semplificata.

### 3.1.2 La generazione della base di conoscenza

A partire dalle formule logiche possono essere generate automaticamente le regole per la produzione di effetti giuridici che costituiscono il nucleo della base della conoscenza di IRI.

Ciascuna di tali regole è la riformulazione di una norma elementare ed ha la seguente sintassi:

**<identificatore della formula >, <numero>,  
<conseguente>, <antecedenti>**

dove

**-<identificatore della formula>** è l'identificatore della formula logica dalla quale la regola è stata estratta;

**-<numero>** è un numero progressivo che distingue ogni regola dalle altre estratte dalla stessa formula;

---

<sup>13</sup>Con IRINORM, ci siano proposti di sviluppare in breve tempo un prototipo funzionante, che potesse rappresentare il punto di partenza per ulteriori sviluppi nel quadro del progetto Irnerio. Molti problemi della rappresentazione della conoscenza giuridica non sono neppure stati affrontati (ad esempio, il problema della rappresentazione delle relazioni concettuali), per altri (ad esempio, il trattamento della negazione) si sono trovate solamente soluzioni provvisorie, non completamente soddisfacenti. Sul problema dei limiti nei quali la conoscenza giuridica può essere rappresentata in clausole di Horn, cfr. cap. 8 par. 6.

<sup>14</sup>Sul problema del passaggio dall'intera logica dei predicati ad un formalismo più ristretto, cfr., tra tutti, KOWALSKI R., *Logic for Problem Solving*, cit. , p. 193 ss.

-<conseguente> è un enunciato atomico (rappresentato nella sintassi presentata al punto precedente) o la negazione di un enunciato atomico (aperto nelle norme generali);

-<antecedenti> è una lista di enunciati atomici o di loro negazioni.

Ad esempio, le due regole riportate nella fig. 8 corrispondono al testo normalizzato della fig. 5.

Se la norma è categorica la lista degli antecedenti è vuota. Con la redazione delle regole si completa la preparazione di una base di conoscenza del sistema. La base di conoscenza consta pertanto di due componenti:

- a. i predicati (cfr. n. 1.2.1),
- b. le regole per la produzione di effetti giuridici di cui si è appena parlato.

**Fig. 8: regole**

```
re(cc.2043,1,r7[P1,P2,D,F],[n(r6[F]),r1[F,P2,D],r2[P1,F],
r3[F]),r5[F,P1,P2]])
re(cc.2043,2,r7[P1,P2,D,F],[n(r6,[F]),r1[F,P2,D],r2[P1,F],
r4[F]),
r5[F,P1,P2]])
```

#### **4. L'ambiente di consultazione**

Nell'ambiente di consultazione di IRINORM è possibile:

- interrogare il sistema, proponendo quesiti concernenti situazioni giuridiche,
- esaminare la base di conoscenza e ripercorrere il processo inferenziale mediante il quale il sistema è giunto a determinate conclusioni.

##### **4.1 L'interrogazione del sistema**

Vengono presentati tutti i possibili quesiti da rivolgere al sistema. Si tratta di tutte le conseguenze delle norme elementari rappresentate nella base della conoscenza che IRINORM sta usando<sup>15</sup> L'utente è chiamato così a specificare l'"effetto giuridico" al quale è interessato.

---

<sup>15</sup>Si stanno sviluppando strumenti che consentano di selezionare solamente le domande attinenti al particolare problema al quale l'utente è interessato.

Se l'enunciato prescelto è la conseguenza di una norma generale (se si tratta cioè di un enunciato aperto, contenente variabili libere) l'utente è può proporre dei termine da sostituire alle (più esattamente, da unificare con i termini che comprendono le) variabili che compaiono nell'enunciato. Per ora è possibile usare solo costanti, e pertanto, se tutte le variabili vengono istanziate si passa necessariamente ad un enunciato chiuso, che descrive una particolare situazione corrispondente al tipo di situazione descritta dall'enunciato prescelto. Ad esempio, se il quesito prescelto era il seguente:

**"E' OBBLIGATORIO CHE**  
**[la persona {P<sup>1</sup>} risarcisce la persona {P<sup>2</sup>} per il**  
**danno {D}] ?"**

si potrà sostituire alla variabile P<sup>1</sup> il nome "Antonio", alla variabile P<sup>2</sup> il nome "Marco", alla variabile "D", l'espressione "distruzione archivi informatici" ottenendo l'enunciato:

**"E' OBBLIGATORIO CHE**  
**[r7 : la persona {antonio} risarcisce la persona {marco}**  
**per il danno {distruzione degli archivi**  
**informatici}],**

enunciato che il sistema cercherà di dimostrare nella prosecuzione dell'interazione con l'utente.

Il sistema chiede se sussistano o meno tutti gli elementi della fattispecie, cioè i fatti che condizionano l'effetto giuridico descritto dall'enunciato (che ne rappresentano la condizione sufficiente).

Per ciascun fatto rispetto al quale l'utente sia interrogato, egli può affermare che il fatto si è verificato, che non si è verificato, o che si ignora se esso si sia verificato o no. Se il fatto in questione è una situazione giuridica<sup>16</sup> il sistema cercherà di dimostrare

---

<sup>16</sup>Cioè se è descritto da un enunciato che costituisce la (o è una specificazione della) conseguenza di una delle norme elementari rappresentate nella base di conoscenza del sistema. Ad esempio, l'enunciato:

**"il fatto {distruzione} da parte di {P1} è ingiusto verso {P2}"**  
 rappresenta una specificazione dell'enunciato

**"il fatto {F} da parte di {P1} è ingiusto verso {P2}"**

che è la conseguenza della seguente norma elementare:

1. [r5 : il fatto {F} da parte di {P1} è ' ingiusto verso {P2}]

**SE**

2. A. **NON SI VERIFICA CHE**

1. [r8:{F} è arrecato nell'esercizio di un diritto di {P1 }] **E**

quel fatto, considerando le condizioni dalle quali dipende la sua realizzazione, in base a quelle regole. Se tutte le condizioni richieste da una regola sussistono (sono affermate dall'utente o dedotte grazie ad altre regole) allora l'elemento della fattispecie viene considerato dimostrato e si passa ad esaminare i restanti elementi. La dimostrazione si svolge mediante un'esame in profondità (depth-first) dell'albero che ha per radice il quesito proposto e nel quale le condizioni che una regola connette ad una conseguenza sono i diretti successori di quest'ultima. In questa ricerca all'indietro (backward) si passa all'esame di un enunciato solo quando né l'enunciato né la sua negazione siano già stati dimostrati.

Qualora l'interrogazione del sistema dia risposta affermativa al quesito proposto, il sistema richiede all'utente se sia interessato a controllare se si possa rispondere al quesito usando regole diverse, cioè con una deduzione diversa da quella (o da quelle) finora compiute. Ciò può essere particolarmente interessante al fine di esaminare quale sia la via più economica per conseguire un certo risultato giuridico.

Qualora invece non sia stato possibile dare una risposta positiva al quesito (non sussistono le condizioni previste da alcuna delle norme con quella conseguenza giuridica presenti nel sistema) e vi sia una regola che preveda come conseguenza giuridica l'inesistenza della situazione giuridica oggetto del precedente quesito, il sistema richiede all'utente se sia interessato a sapere se si possa escludere che quella situazione si sia verificata.

Infatti, la mancanza della prova di un certo enunciato (la sua non-deducibilità dalle premesse di cui il sistema dispone), non equivale nel nostro sistema alla prova della negazione di quell'enunciato.

## 4.2 Il controllo della deduzione e della base di conoscenza

La procedura per il controllo della base della conoscenza consente di

- esaminare la base della conoscenza a lungo termine, cioè le conoscenze giuridiche di cui il sistema dispone;
- controllare il processo inferenziale seguito per rispondere al quesito proposto. Per una spiegazione, seppur rudimentale, delle conclusioni cui si è giunti, è possibile ripercorrere il processo inferenziale esaminando le regole grazie alle quali si è risposto al quesito, nell'ordine in cui sono state utilizzate;

---

B. 1. [r9:{F}] è lesione di interesse tutelato giuridicamente di {P2}]  
 regola che può essere estratta dalla formula "ingiustizia" (cfr. fig. 4).

-esaminare se dai fatti affermati dall'utente discendano conseguenze ulteriori, oltre a quelle dedotte per rispondere al quesito, ragionando in avanti (*forward*) a partire da quei fatti.



1. La struttura del sistema	285
2. Principi ispiratori	286
2.1 La tecnica della normalizzazione	286
2.2 L'uso della logica per esprimere contenuti giuridici	291
3. L'ambiente di sviluppo	293
3.1 L'inserimento e la modifica della conoscenza	293
4. L'ambiente di consultazione	301
4.1 L'interrogazione del sistema	301
4.2 Il controllo della deduzione e della base di conoscenza	303



## BIBLIOGRAFIA

### Elenco delle opere citate

- AARNIO A., ALEXI R., PECZENIK A., *Grundlagen der juristischen Argumentation*, in KRAVIETZ W., ALEXI R., *Metatheorie juristischer Argumentation*, pp. 9-87.
- AARNIO A., *On Changes on the Systematics of Law*, in ECKHOFF T., FRIEDMAN L.M., UUSITALO J. (a cura di), *Vernunft und Erfahrung in Rechtsdenken der Gegenwart*, 1986, pp. 161-170.
- AAVV, *La redazione di testi legislativi: tecniche e strumenti nuovi*, ISAS, Palermo, 1986.
- AAVV, *Thesaurus del linguaggio legislativo*, ISAS, Palermo, 1986.
- AGAZZI E., *Introduzione ai problemi dell'assiomatica, Vita e pensiero*, Milano, 1961.
- AIT-KACI H., NASR R., *Login: A Logic Programming Language with Built-in Inheritance*, in "Journal of Logic Programming", 3, 1986, pp. 185-205.
- ALCHOURRON C., BULYGIN E., *Normative Systems*, Springer Verlag, Wien, New York, 1971.
- ALCHOURRON C., *Conditionality and the Representation of Legal Norms*, in MARTINO A.A., SOCCI NATALI F. (a cura di) *Automated Analysis of Legal Texts*, 1986, pp. 175-186.
- ALCHOURRON C., *Systematisation and Change in the Science of Law*, in ECKHOFF T., FRIEDMAN L.M., UUSITALO J. (a cura di), *Vernunft und Erfahrung in Rechtsdenken der Gegenwart*, 1986, pp. 171-184.
- ALEXI R., *Theorie der juristische Argumentation*, Suhrkamp, Frankfurt, 1978.
- ALLEN L.E., *Formalization of Hohfeldian Analysis to Clarify the Multiple Senses of 'Legal Right': A Powerful Lens for The Electronic Age*, in "Southern California Law Review", 1974, pp. 428-487.
- ALLEN L.E., *Language, Law and Logic: Plain Legal Drafting for the Electronic Age*, in NIBLETT (a cura di) *Computer Science and Law: An advanced Course*, 1986, pp. 75-100.

- ALLEN L.E., *Symbolic Logic. A Razor-Edged Tool for Drafting and Interpreting Legal Documents*, in "Yale Law Journal", 66, 1957, pp. 833-879.
- ALLEN L.E., *Towards a Normalized Language to Clarify the Structure of Legal Discourse*, in MARTINO A.A. (a cura di), *Deontic Logic, Computational Linguistics and Legal Informations Systems*, 1982, pp. 349-407.
- ALLEN L.E., *Una guida per redattori giuridici di testi normalizzati*, in "Informatica e diritto", 2, 1978, pp. 61-114.
- ALLEN L.E., SAXON C.S., *Analysis of the Logical Structure of Legal Rules by a Modernized and Formalized Version of Hohfeld's Fundamental Legal Conceptions*, in MARTINO A.A., SOCCI NATALI F. (a cura di) *Automated Analysis of Legal Texts*, 1986, pp. 385-450.
- ALLEN L.E., SAXON C.S., *Automatic generation of a legal expert system of section 7 (2) of the United Kingdom data protection act 1984*, Relazione presentata al 4° Congresso internazionale sul tema *Informatica e regolamentazioni giuridiche* (Roma, maggio 1988).
- ALLEN L.E., SAXON C.S., *Computer Aided Normalizing and Unpacking: some Interesting Machine-processable Transformations of Legal Rules*, in WALTER C. (a cura di), *Computing Power and Legal Reasoning*, 1985, pp. 495-572.
- ALLEN L.E., SAXON C.S., PAYTON S.A., *Syntesizing related Rules from Statutes and Cases for Legal Expert Systems*, relazione presentata al Congresso internazionale *Expert Systems in Law*, (Bologna, maggio 1989).
- ALLEN L.E., SAXON C.S., *Some problems in designing expert sistems to aid legal reasoning*, in *Proceedings of the First International Conference on Artificial Intelligence and Law* (Boston, maggio 1987), 1987, pp. 94-103.
- ALT F.L., RUBINOFF M., *Advances in Computer*, 6, Academic Press, New York, 1965.
- ALTHY J.L., COOMBS M.J., *Expert Systems. Concepts and Examples*, The National Computing Centre, Manchester, 1984.
- ANDERSON A.R., BELNAP N., *Entailment*, Princeton University Press, Princeton, 1975.
- ANDRETTA M., LUGARESI M., ZAMBON F., LOSANO M., NANNINI N., *I linguaggi formali applicati alla rappresentazione di testi normativi: il progetto PROLEG (PROlog applicato alle LEGgi)*, Relazione

presentata al 41 Congresso internazionale sul tema *Informatica e regolamentazioni giuridiche* (Roma, maggio 1988).

- ANDRETTA M., LUGARESI M., ZAMBON F., LOSANO M., NANNINI N., *Uso di linguaggi formali per la rappresentazione di testi normativi: il progetto PROLEG (PROlog applicato alle LEGgi)*, Atti del terzo convegno nazionale sulla programmazione logica, 1988, pp. 373-383.
- AQUIST L., *Deontic Logic*, in GABBAY D.M., GUENTHNER F. (a cura di), *Handbook of Philosophical Logic. Vol. 2. Extensions of Classical Logic*, Reidel, Dordrecht, 1984, pp. 605-714.
- ARISTOTELE, *La Politica*, Laterza, Bari, 1966, pp. 421.
- ARNAUD A.J., *Dictionnaire encyclopedique du théorie du droit et de sociologie du droit*, Librairie de droit ed de Jurisprudence, Paris, Story-Scientia, Bruxelles, 1988, pp. 487.
- ARNOLD C., *Yearbook of Law, Computers and Technology*, Vol. 2, Butterworth, London, 1986.
- ASHLEY K.D., *Reasoning by Analogy: A Survey of Selected AI Research with Implications for Legal Expert Systems*, in WALTER C. (a cura di), *Computing Power and Legal Reasoning*, 1985, pp. 105-127.
- ASHLEY K.D., RISSLAND E.L., *But, See, Accord. Generating 'Blue Book' Citations in HYPO*, in *Proceedings of the First International Conference on Artificial Intelligence and Law* (Boston, maggio 1987), 1987, pp. 67-74.
- ASHLEY K.D., RISSLAND E.L., *Toward Modelling Legal Arguments*, in MARTINO A.A., SOCCI NATALI F. (a cura di) *Automated Analysis of Legal Texts*, 1986, pp. 19-39.
- BAADE H.W. (a cura di), *Jurimetrics*, Basic Books, New York, 1963 (originariamente pubblicato in *Law Contemp. Probl.*, 28, 1963, pp. 1-270).
- BARR A., FEIGENBAUM E.A., COHEN P. R. (a cura di), *Handbook of Artificial Intelligence*, Kaufmann, Los Altos (California), 1981.
- BARTHEL T., *Structured Programs as a Paradigm of Structured Laws*, in NIBLETT (a cura di), *Computer Science and Law*, 1979, pp. 147-156.
- BECKER O., *Untersuchungen über den Modalkalkül*, A. Hain, Maisenheim am Glam, 1952.

- BELL J.L., MACHOVER M., *A Course in Mathematical Logic*, North Holland, Amsterdam, 1977.
- BELLORD N.J., *Tax Planning by Computer*, in NIBLETT B. (a cura di), *Computer Science and Law: An Advanced Course*, 1979, pp. 173-182.
- BENCH-CAPON T.J.M., ROBINSON G.O., ROUTEN T.W., SERGOT M.J., *Logic Programming for Large Scale Applications in Law: A Formalisation of Supplementary Benefit Legislation*, in *Proceedings of the First international Conference on Artificial Intelligence and Law* (Boston, Maggio 1987), 1987, pp. 190-198.
- BENCIVENGA E., *Introduzione ai metodi della logica contemporanea*, Boringhieri, Torino, 1984.
- BERTOCCHI R., FRATTINI F., LANZARONE G.A., MAGLIA A., *Programmazione in logica della legge sull'IVA*, in *Atti del terzo Convegno nazionale sulla Programmazione logica*, 1988, pp. 355-371.
- BERTOCCHI R., *Rappresentazione di testi di leggi in PROLOG*, in *"Informatica e diritto"*, 3, 1986, pp. 155-167.
- BIAGIOLI C., BIAGINI P., DINI G., MARTINO A.A., SOCCI F., TISCORNIA D., TRIVISONNO G., *An Arithmetic Model for Analysing Normative Systems: Some Experimental Proposals*, Istituto per la documentazione giuridica del Consiglio nazionale delle ricerche, Firenze, giugno 1982.
- BIAGIOLI C., FAMELI E., *Expert Systems in Law: An international survey and a selected bibliography*. Istituto per la documentazione giuridica, Firenze, in *"CCAI"* (the Journal for the Integrated Study of Artificial Intelligence, Cognitive Science and Applied Epistemology) 4, 3, 1987.
- BIAGIOLI C., MARIANI P., TISCORNIA D., *ESPLEX: A rule based and conceptual based model for representing statutes*, in *Proceedings of the First international Conference on Artificial Intelligence and Law* (Boston, Maggio 1987), 1987, pp. 240-251.
- BIGELOW R.P. (a cura di), *Computers and the Law. An Introductory Handbook*, American Bar Association, Commerce Clearing House, Chicago (Illinois), 1981 (terza edizione).
- BILON J.L., *A Knowledge Base of Reasoning and Judgements*, relazione presentata al Congresso internazionale *Expert Systems in Law*, (Bologna, maggio 1989).

- BING J., *Designing Text Retrieval Systems for 'Conceptual Searching'*, in *Proceedings of the First International Conference on Artificial Intelligence and Law* (Boston, maggio 1987), 1987, pp. 43-51.
- BLAZER A. (a cura di), *Natural Language and the Computer, Scientific Symposium on Syntax and Semantics for Text Processing and Man-Machine-Communication Held on Occasion of the 20th Anniversary of the Science Center Heidelberg of IBM Germany. Proceedings* (Heidelberg, febbraio 1988), Springer, Berlin, 1988.
- BOBBIO N., *Scienza del diritto e analisi del linguaggio*, in "Rivista trimestrale di diritto e procedura civile", 1950, pp. 342-367
- BOBBIO N., *Teoria della norma giuridica*, Giappichelli, Torino, 1958.
- BODARD F., HELLA M., POULLET Y., STENNE P., *A Prototype ADP System to Assist Judicial Decision Making*, in MARTINO A.A., SOCCI NATALI F. (a cura di) *Automated Analysis of Legal Texts*, 1986, pp. 187-210.
- BOURCIER D., *Ces systèmes dits experts ou comment passer du droit à la decision juridique*, 1987, Relazione presentata al Congresso Informatique et Droit (Strasburgo, novembre 1987).
- BRACHMAN, SMITH (a cura di), *SIGART Newsletter: Special Issue on Knowledge Representation*, Volume 70, 1980.
- BUCHANAN B.G., FEIGENBAUM E.A., *Dendral and Metadendral: their application dimension*, in "Artificial Intelligence", 2, 1978, pp. 5-24.
- BUCHANAN B.G., HEADRICK T.E., *Some Speculations about Artificial Intelligence and Legal Reasoning*, in "Stanford Law Review", 23, 1970, pp. 40-62.
- BUCHANAN B.G., SHORTLIFFE E., *Rule-Based Expert Systems*, Addison-Wesley, Reading (Massachusetts), 1984.
- BULL R.A., SEGERBERG K., *Basic Modal Logics*, in GABBAY D.M., GUENTHNER F. (a cura di), *Handbook of Philosophical Logic. Vol. 2. Extensions of Classical Logic*, Reidel, Dordrecht, 1984, pp. 1-88.
- CAMELLI, SOCCI NATALI F., *Strumenti automatici nel diritto. Lexis: un progetto di sistema esperto giuridico in xi plus*, Relazione presentata al 41 congresso internazionale sul tema Informatica e regolamentazioni giuridiche (Roma, maggio 1988).
- CAMPBELL (a cura di), *Data Processing and the Law*, Sweet & Maxwell, London, 1984.

- CARIDI G., PELLECCCHIA S., *Automazione della ricerca giuridica e sistemi esperti*, Angeli, Milano, 1986.
- CASADEI G., TEOLIS A., *Introduzione all'informatica. La programmazione*, Zanichelli, Bologna, 1979.
- CASADEI G., TEOLIS A., *Prolog. Dalla programmazione all'intelligenza artificiale*, Bologna, Zanichelli, 1986.
- CECCATO S., *Automatic Translation of Languages*, in "Inf. Stor. Retr.", 2 (3), 1964, pp. 105-158.
- CECCATO S., *Linguistic Analysis and Programming for Mechanical Translation*, Gordon and Breach, New York, 1961.
- CHELLAS B.F., *Modal Logic. An Introduction*, Cambridge University Press, London, 1980.
- CHOMSKI N., *Aspects of the Theory of Syntax*, M.I.T. Press, Cambridge (Massachusetts), 1965 (trad. it. *Aspetti della teoria della sintassi*, in *Saggi linguistici. 2. La grammatica generativa trasformativa*, 1979, pp. 41-258).
- CHOMSKI N., *Saggi linguistici. 2. La grammatica generativa trasformativa*, Boringhieri, Torino, pp. 352, 1970 (seconda edizione 1979),
- CHOMSKI N., *Syntactic Structures*, Mouton, The Hague, 1957, pp. 118 (ed. 12: 1976) (trad. it. *Le strutture della sintassi*, Laterza, Bari, 1970).
- CIAMPI C. (a cura di), *Artificial Intelligence and Legal Information Systems*, North Holland, Amsterdam, 1982.
- CIAMPI C., *LABEO: A Knowledge-based Expert System for the <<Animation>> of Legal Texts*, in "Informatica e diritto", 3, 1988, pp. 183-202.
- CIAMPI C., *Labeo: un sistema esperto per la rappresentazione e l'elaborazione della conoscenza giuridica*, in AAVV *La redazione dei testi legislativi: tecniche e strumenti nuovi*, 1986, pp. 61-83
- CIAMPI C., SARTOR G., *Un thesaurus di descrittori per la legge sul condono edilizio*, in AAVV, *Thesaurus del linguaggio legislativo*, 1986, pp. 1-22.
- CLARK K.L., *Negation as Failure*, in GALLAIRE H., MINKER J. (a cura di) *Logic and Data Bases*, 1978, pp. 293-322



- CLARK K.L., TARNLUND S.A. (a cura di), *Logic Programming*, Academic Press, London-New York, 1982.
- CLOCKSIN W.F., MELLISH C.S., *Programming in Prolog*, Springer Verlag, New York, 1981.
- COLMERAUER A., KANOUI H., PASERO R., ROUSSEL P., *Un système de communication homme-machine en Français, relazione interna, Groupe d'intelligence artificielle, Université d'Aix-Marseille, Luminy, France, 1972.*
- CONTE A., *Deontico vs. dianoetico*, in "Materiali per una storia della cultura giuridica", 1986, pp. 489-494
- COPI I., *Introduction to logic*, Mac Millan, New York, 1961 (trad. it. *Introduzione alla logica*, Il Mulino, Bologna, 1964)
- CROSS G.R., DE BESSONET C., *Representation of Some Aspects of Legal Causality*, in WALTER C. (a cura di), *Computing power and legal reasoning*, 1985, pp. 205-214.
- CROSS G.R., DE BESSONET C.G., *Conceptual Retrieval and Legal Decision Making*, in MARTINO A.A., SOCCI NATALI F. (a cura di) *Automated Analysis of Legal Texts*, 1986, pp. 219-227
- CROSS G.R., DE BESSONET C.G., *Legislative Techniques and Automated Systems*, in *Atti del terzo congresso su l'informatica giuridica e la comunita nazionale ed internazionale*, Roma, 1983
- D'AIETTI G., *Un expert system in tema di locazioni commerciali*, Relazione presentata al 41 Congresso internazionale sul tema *Informatica e regolamentazioni giuridiche* (Roma, maggio 1988).
- DAVIDSON D., HARMANN G., *Semantics of Natural languages*, Reidel, Dordrecht, 1972.
- DEGANO P., SANDEWALL E. (a cura di), *Integrated Interactive Computer Systems*, North-Holland, Amsterdam, 1983.
- DEGLI ANTONI G., ZONTA B., *Analysis of Laws by means of Petri Nets: Motivations and Metodology*, in CIAMPI (a cura di) *Artificial Intelligence and Legal Information Systems*, 1982, pp. 273-299.
- DI BERNARDO G., *Logica, norme, azione*, Istituto Superiore di Scienze Sociali, Trento, 1969.
- DIJKSTRA E.W., *A Discipline of Programming*, Prentice-Hall, Englewood Cliffs (NJ), 1976.
- DREYFUS H.L., DREYFUS S.E., *Mind over Machine*, Blackwell, Oxford, 1986.

- DWORKIN R.M., *Taking Rights Seriously*, Duckworth, London, 1977.
- ECKHOFF T., FRIEDMAN L.M., UUSITALO J. (a cura di), *Vernunft und Erfahrung in Rechtsdenken der Gegenwart*, "Rechtstheorie", Beiheft 10, Duckner & Humblot, Berlin, 1986.
- ERDMANN U., FIEDLER H., HAFT F., TRAUNMÜLLER R. (cura di), *Computergestützte juristische Expertensysteme*, Attempto Verlag, Tübingen, 1986.
- FAHRENBACH H. (a cura di) *Wirklichkeit und Reflexion*, Festschrift für W. Schulz, Pfullingen, 1973.
- FAMELI E., *L'automa infortunistico: un esperimento di consulenza giuridica automatica*, in "Informatica e diritto", 1, 1976, pp. 1-50.
- FAMELI E., NANNUCCI R., *I sistemi esperti nel diritto. Strumenti e metodi di sviluppo*, Relazione presentata al 41 congresso internazionale sul tema *Informatica e regolamentazioni giuridiche* (Roma, maggio 1988).
- FASSO' G., *Storia della filosofia del diritto. Volume III. Ottocento e novecento*, Il Mulino, Bologna, 1970.
- FEIGENBAUM E.A., FELDMAN J. (a cura di), *Computers and Thought*, Mc Graw-Hill, New York, 1963.
- FEYS R., *Les logiques nouvelles des modalités*, in "Revue Néoscholastique de Philosophie", vol. 40, 1937, pp. 517-553.
- FEYS R., *Les logiques nouvelles des modalités*, in "Revue Néoscholastique de Philosophie", vol. 41, 1938, pp. 217-252.
- FIEDLER H., BARTHEL T., VOOGD G., *Formalisierung im Recht. Untersuchungen zur formalisierung Recht als Beitrag zur Grundlageforschung juristischer Datenverarbeitung*, Westdeutscher Verlag, Leverkusen, 1984.
- FIEDLER H., *Expert Systems as a Tool for Drafting Legal Decisions*, in MARTINO A.A., SOCCI NATALI F. (a cura di) *Automated Analysis of Legal Texts*, 1986, 607-612.
- FIEDLER H., *Structured Programs as a Paradigm of Structured Laws*, in NIBLETT (a cura di), *Computer Science and Law*, 1979, pp. 137-146.
- GABBAY D.M., GUENTHNER F. (a cura di), *Handbook of Philosophical Logic. Vol. 1. Elements of Classical Logic*, Reidel, Dordrecht, 1983.

- GABBAY D.M., GUENTHNER F. (a cura di), *Handbook of Philosophical Logic*. Vol. 2. *Extensions of Classical Logic*, Reidel, Dordrecht, 1984.
- GABBAY D.M., SERGOT M.J., *Negation as Inconcistency. I*, in "Journal of Logic Programming", 1986, 1, pp. 1-35.
- GALGANO F., voce *Negoziio Giuridico* (premesse problematiche e dottrine generali, in *Enciclopedia del diritto*, XXVII, 1977, pp. 932 ss+.
- GALLAIRE H., MINKER J. (a cura di), *Logic and Data Bases*, Plenum Press, New York, 1978.
- GARDNER A., *Law Applications*, in SHAPIRO S.C., ECKROTH D., *Encyclopedia of artificial intelligence*, 1987, pp. 456-463.
- GENZEN G., *Untersuchungen über dal logischen Schliessen*, "Mathematische Zeitschrift", 39, 1934, pp. 176-210, pp. 405-431.
- GÖDEL K., *Collected works*, Oxford University Press, New York, 1986, pp. 301-302 (già pubblicato come *Eine Interpretation des Intuitionistischen Aussagenkalkulus*, in *Ergebnisse eines Mathematischen Kolloquiums*, 4, 1933, pp. 34-38).
- GÖDEL K., *Über formal unentschiedbare Sätze der Principia Mathematica und verwandter Systeme*, in "Monatshefte für Matematik und Phisik", 39, 1931 (trad. it. in in AGAZZI E., *Introduzione ai problemi dell'assiomatica*, Vita e pensiero, Milano, 1961),
- GOLD D.L., SUSSKIND R.E., *Expert Systems in Law: A Jurisprudential and Formal Specification Approach*, in MARTINO A.A., SOCCI NATALI F. (a cura di) *Automated Analysis of Legal Texts*, 1986, pp. 625-642.
- GOOD I.J., *Speculations concerning the first ultraintelligent machine*, in ALT F.L., RUBINOFF M., *Advances in Computer*, 6, Academic Press, New York, 1965, pp. 31-38.
- GORDON T.F., *Object-Oriented Predicate Logic and Its Role in Representing Legal Knowledge*, in WALTER C. (a cura di), *Computing Power and Legal Reasoning*, 1985, pp. 163-203.
- GORDON T.F., QUIRCHMAYR G., *Ein hybrides Wissensrepräsentationssystem zur Modellierung rechtswissenschaftlicher Probleme*, in 16. Jahrestagung der Gesellschaft für Informatik, Berlin, 1986.
- GORDON T.F., QUIRCHMAYR G., OBLOG. *Eine Programmiersprache für juristische Expertensysteme*, in ERDMANN U., FIEDLER H., HAFT F., TRAUNMÜLLER R. (cura di), *Computergestützte juristischer Expertensysteme*, 1986, pp. 123-134.

- GRAY G., *Statutes Enacted in Normalized Form: the Legislative Experience in Tennessee*, in WALTER C. (a cura di), *Computing Power and Legal Reasoning*, 1985, pp. 621-626
- GREENLEAF G., MOMBRAY A., TYREE A.L., *Expert Systems in Law: The DATALEX Project*, in *Proceedings of the First International Conference on Artificial Intelligence and Law* (Boston, maggio 1987), 1987, pp. 9-17.
- GROENENDIJK J.A.G., JANSSEN T.M.V., STOKHOF M.B.G., *Formal Methods in the Study of Language*, MC TRACT 135, Eds. Amsterdam, Amsterdam, 1981.
- GUENTHER F., LEHMANN H., *Automatic Costruction of Discourse Representatio Structures*, in *Proceedings of the 10th International Conference on Computational Linguistics*, Stanford (CA), pp. 398 ss., 1984.
- GÜNTHER F., LEHMANN H., SCHÖNFELD W., *A Theory for the Representation of Knowledge*, in "IBM Journal of Research and Development", vol. 30, n. 1, gennuary, 1986, pp. 39-56.
- HAACK S., *Philosophy of Logics*, Cambridge University Press, Cambridge, 1978 (trad. it. HAACK S., *Filosofia delle logiche*, Franco Angeli, Milano, 1983).
- HABERMAS J., *Theorie des kommunikativen Handelns*, Suhrkamp, Frankfurt, 1981.
- HABERMAS J., *Wahrheitstheorien*, in FAHRENBACH H. (a cura di) *Wirklichkeit und Reflexion, Festschrift für W. Schulz*, 1973, pp. 211-265.
- HAFNER C.D., *An Information Retrieval System based on a Computer Model of Legal Knowledge*, UMI Research Press, Ann Arbor (Michigan), 1981.
- HAFNER C.D., *Rapresentation of Knowledge in a Legal Information Retrieval System*, in ODDY S., ROBERTSON C., VAN RIJSBERGEN C., WILLIAMS P., *Information Retrieval Research*, 1981.
- HALL H., *Phenomenology*, in SHAPIRO S.C., ECKROTH D., *Encyclopedia of Artificial Intelligence*, 1987, pp. 730-736
- HAMMOND P., SERGOT M.J., APES. *Augmented Prolog for Expert Systems. System Documentation*, Logic Based Systems LTD., 1987 (terza edizione)
- HART H.L.A., *The Concept of Law*, Oxford University Press, London, 1961 (Trad. it. *Il concetto di diritto*, Einaudi, 1965)

- HAYES-ROTH F., *Rule-based Systems*, in SHAPIRO S.C., ECKROTH D., *Encyclopedia of Artificial Intelligence*, 1987, pp. 963-973.
- HEATHER M.A., *Future Generation Computer Systems*, in MARTINO A.A., SOCCI NATALI F. (a cura di) *Automated Analysis of Legal Texts*, 1986, pp. 643-660.
- HELLAWELL R., *A computer program for legal planning and analysis: Taxation of stock redemptions*, in "Columbia Law Review" 80, 7, 1980, pp. 1363-1398.
- HELLAWELL R., *CHOOSE: A computer program for legal planning and analysis*, in "Columbia Journal of Transnational Law", 19, 1981, pp. 339-357.
- HELLAWELL R., *SEARCH: A Computer program for legal problem solving*, in "Akron Law Review", 15, 1982, pp. 635-653.
- HILPINEN R. (a cura di), *New Studies in Deontic Logic*, Riedel, Dordrecht, 1981.
- HODGES W., *Elementary Predicate Logic*, GABBAY D.M., GUENTHNER F. (a cura di), *Handbook of Philosophical Logic. Vol. 1. Elements of Classical Logic*, Reidel, Dordrecht, 1983, pp. 1-131.
- HODGES W., *Introduction to Elementary Logic*, Penguin Books, Harmondsworth, Middlessex, England, 1977 (trad. it. *Logica*, Garzanti, Milano, 1986).
- HOFSTADTER D.R., DENNET D.C., *The Mind's I*, Basic Books, New York, 1981.
- HOGGER J.C., *Introduction to Logic Programming*, Academic Press, London, 1984.
- HOHFELD W.N., *Concetti giuridici fondamentali*, Einaudi, Torino, 1969.
- HOLMES O.W., *The Path of the Law*, in *Collected Legal Papers*, New York, 1920.
- HUGHES G.E., CRESSWELL M.J., *An Introduction to Modal Logic*, Methuen, London, 1968 (ristampa con correzioni 1972, ristampa 1982).
- JONES A.J., *On the Relationship between Permission and Obligation*, in *Proceedings of the First International Conference on Artificial Intelligence and Law* (Boston, maggio 1987), 1987, pp. 164-169.
- JONES R., MASON P., STAMPER R., *Legol 2.0: A Relational Specification Language for Complex Rules*, in "Information Systems", 4, 1979, pp. 293-305.

- KALINOWSKI G., *Théorie des propositions normatives in Logique déontique. I* (1953-1969), 1972, pp. 147-182 (già in "Studia logica", 1953).
- KALINOWSKI G., *Etudes de logique déontique. I* (1953-1969), Librairie générale de droit et de jurisprudence, Paris, 1972.
- KALISH D., MONTAGUE R., *Logic. Techniques of Formal Reasoning*, Harcourt, Brace & World, New York, 1964.
- KAMP H., *A Theory of Truth and Semantic Representation*, in GROENENDIJK J.A.G., JANSSEN T.M.V., STOKHOF M.B.G., *Formal Methods in the Study of Language*, MC TRACT 135, Eds. Amsterdam, 1981, pp. 277-322.
- KAMP H., *Discourse Representation Theory: What it is and where it ought to go*, in BLAZER A. (a cura di) *Natural Language and the Computer*, 1988, pp. 84-111.
- KEEN M., MCBRIDE S., *Expert systems for Clarifying Employment Law*, in *Knowledge Based System 86*, Online, Pinner, 1986.
- KELSEN H., *Allgemeine Theorie der Normen*, Manzsche Verlags und Universitätsbuchhandlung, Wien, 1979 (trad.it. *Teoria generale delle norme*, Einaudi, Torino, 1985)
- KELSEN H., *Reine Rechtslehre*, Verlag Franz Deuticke, Wien, 1960 (trad. it. *Dottrina pura del diritto*, Einaudi, Torino, 1975)
- KING J.J., *Analysis of a KRL Implementation of a Current Legal Reasoning Program Design* (non pubblicato), 1976.
- KLAHR P., WATERMAN D.A., *Expert Systems. Techniques, Tools and Applications*, Addison-Wesley, Reading (Massachusetts), +, pp. 135-185.
- KLUG U., 1951 (3a ed 1963), *Juristische Logik*, Springer, Berlin.
- KNUTH T.S., *The Art of Computer Programming*, Addison, Wesley, Reading (Massachusetts), 1973 (21 ediz.).
- KOLATA G., *How can Computers get Common Sense*, in "Science", 217, 1987, pp. 1237 ss.
- KORNHAUSER L.A., *Economique (Analyse -du droit)*, in *Dictionnaire encyclopedèdique du théorie du droit et de sociologie du droit* (diretto da ARNAUD A.J.), Librairie de droit ed de Jurisprudence, Paris, Story-Scientia, Bruxelles, 1988, pp. 129-130.

- KOWALSKI R.A., in BRACHMAN R.J., SMITH B.C. (a cura di), *SIGART Newsletter: Special Issue on Knowledge Representation*, 1980.
- KOWALSKI R.A., *Logic for Problem Solving*, North Holland, New York, Oxford, 1979.
- KOWALSKI R.A., *Logic Programming*, in SHAPIRO S.C., ECKROTH D. (a cura di), *Encyclopedia of Artificial Intelligence*, 1987, pp. 544-558.
- KOWALSKI R.A., *Notes on the British Nationality Act*, non pubblicato, 1988.
- KOWALSKI R.A., SERGOT M.J., *The Use of Logical Models in Legal Problem Solving*, relazione presentata al Congresso internazionale *Expert Systems in Law*, Bologna, 1989.
- KOWALSKI R.A., *The Predicate Calculus as a Programming Language*, in *Proceedings of the First Symposium on the Mathematical Foundations of Computer Science*, Jablonna (Poland), 1972.
- KRAVIETZ W., ALEXI R., *Metatheorie juristischer Argumentation*, Duncker & Humblot, Berlin, 1983.
- KRAVIETZ W., SCHELSKI H., WINKLER G., SCHRAMM A. (a cura di), *Theorie der Normen. Festgabe für Ota Weinberger zum 65 Geburtstag*, Duncker & Humblot, Berlin, 1984.
- KRIPKE S., *Naming and Necessity*, in DAVIDSON D., HARMANN G., *Semantics of Natural Languages*, Dordrecht, Reidel, 1972, pp. 253-355.
- KUHN T., *The Structure of Scientific Revolution*, Chicago, University of Chicago Press, 1962 (trad. it. *La struttura della rivoluzioni scientifiche*, Einaudi, Torino, 1969).
- LANZARONE G.A., *Algoritmo*, in *Gli strumenti del sapere contemporaneo, Volume II, I concetti*, Grande dizionario enciclopedico, Utet, Torino, 1985, pp. 7-9.
- LANZARONE G.A., *Informatica*, in *Gli strumenti del sapere contemporaneo, Volume I, Le Discipline*, Grande dizionario enciclopedico, Utet, Torino, 1985, pp. 384-400.
- LANZARONE G.A., MAIOCCHI R., POLILLO R., *Introduzione alla programmazione strutturata. Il caso del Fortran, Cobol e Assembler*, Franco Angeli, Milano, 1981.
- LEHMANN H., *Formale Răpresentation juristisches Wissens*, ERDMANN, FIEDLER, HAFT, TRAUMMÜLLER (a cura di), *Computergestützte juristischer Expertsysteme*, 1986, pp. 75-86.

- LEHMANN H., 1988, *The LEX-Project - Concepts and Results*, in BLAZER A. (a cura di), *Natural Language and the Computer*, 1988, pp. 112-146.
- LEHMANN H., *Das juristische Konsultationssystem Lex aus software-technischer Sicht*, ERDMANN U., FIEDLER H., HAFT F., TRAUNMÜLLER R. (cura di), *Computergestützte juristischer Expertsysteme*, 1986, pp. 49-74.
- LEHMANN H., *Legal Concepts in a Natural Language based Expert system*, relazione presentata al Congresso internazionale Expert Systems in Law (Bologna, maggio 1989).
- LEHMANN H., ZOEPPRITZ M., *Formale Behandlung der Begriffs 'Pflicht'*, in HOMMEL G., SCHINDLER S. (a cura di) *GI Jahrestagung 6.-10. Oktober 1986*, II, 1986, pp. 392-405.
- LEIBNIZ G.W., *Nova methodus docendae discendaeque jurisprudentiae ex artis didacticae principiis in parte generali premissis experientiae luce*, in *Sämtliche Schriften und Briefe*, vol. VI, I, Darmstad, 1930 (1667), pp. 259-634.
- LLOYD J.W., *Foundations of Logic Programming*, Springer, Berlin, 1984 (trad. it. *Fondamenti della programmazione logica*, Muzzio, Padova, 1986).
- LEITH P., *Clear Rules and Legal Expert Systems*, in MARTINO A.A., SOCCI NATALI F., *Automated Analysis of Legal Texts*, 1986, pp. 661-680.
- LEITH P., *ELI: An expert legislative consultant.*, Relazione presentata al Congresso IEE Conference on Man/Machine Systems (UMIST, Manchester, July 1981).
- LEITH P., *Fundamental Errors in Legal Logic Programming*, in "The Computer Journal", 1986, pp. 545-552.
- LEITH P., *Hierarchically structured production rules*, in "The Computer Journal", 1983, pp. 26 ss.
- LEVINE, MINEL, MEYER, *Un systeme expert d'aide au controle de prestations sociales: l'allocation aux adultes handicapés*, Relazione presentata al Congresso Informatique et Droit (Strasburgo, novembre 1987).
- LLOYD M., *Legal Databases in Europe. User Attitudes and Supplier Strategies*, North Holland, Amsterdam, 1986, pp. 218.
- LOEVINGER L., *Jurimetrics: The Next Step Forward*, in "Minnesota Law Review", XXXIII, 1949, p. 455 ss.



- LOSANO M., *Corso di informatica giuridica*, Unicopli, Milano, 1984.
- LOSANO M., *Informatica per le scienze sociali*, Einaudi, Torino, 1985, pp. 547.
- LOSANO M., *L'analisi delle procedure giuridiche*, in *Corso di informatica giuridica*, vol. 3, 1984.
- LOSANO M., *L'informatica e l'analisi delle procedure giuridiche*, Unicopli, Milano, 1989.
- LOSANO M., *Scritto con la luce. Il disco compatto e la nuova editoria elettronica*, Unicopli, Milano, 1988.
- LOVELAND D.W., *Mechanical Theorem Proving by Model Elimination*, in JACM, 15, 1968, pp. 223-251.
- MACKAAY E., POULIN D., FREMONT J., DENIGER C., BRATLEY P., *The Logic of Time in Law and Legal Expert Systems*, relazione presentata al Congresso internazionale Expert Systems in Law (Bologna, maggio 1989).
- MAGGS P.B., DE BESSONET C.G., *Automated Logical Analysis of Systems of Legal Rules*, in "Jurimetrics Journal", 12, 1972, pp. 150-169.
- MALLERY J.C., HURWITZ R., *Hermeneutics*, in SHAPIRO S.C., ECKROTH D., *Encyclopedia of Artificial Intelligence*, 1987, pp. 362-376.
- MANZANARES, NECTOUX, *L'informatique au service du juriste*, Librairies techniques, Paris, 1987.
- MARASCHINI F., *L'accesso intelligente alle basi informative*, Relazione presentata al 41 Congresso internazionale sul tema *Informatica e regolamentazioni giuridiche* (Roma, maggio 1988).
- MARIANI P., TISCORNIA D., *I sistemi esperti di supporto alla redazione di testi legislativi*, Relazione presentata al 41 congresso internazionale sul tema *Informatica e regolamentazioni giuridiche* (Roma, maggio 1988).
- MARTINO A.A. (a cura di), *Deontic Logic, Computational Linguistics and Legal Informations Systems*, New Holland, Amsterdam, 1982.
- MARTINO A.A., ABBA L., ASIRELLI P., CAMMELLI A., MARIANI P., MARTELLI M., SOCCI F., TISCORNIA D., *Knowledge Base in the Automated Analysis of Legislation*, in MARTINO A.A., SOCCI NATALI F., *Automated Analysis of Legal Texts*, 1986, pp. 281-286.
- MARTINO A.A., SOCCI NATALI F. (a cura di), *Automated Analysis of Legal Texts*, North Holland, Amsterdam, 1986.

- MARTINO A.A., *Software per il legislatore*, in "Informatica e diritto", 3, 1987, pp. 25-64.
- MCCARTY T., *Reflections on TAXMAN: An Experiment in Artificial Intelligence and Legal Reasoning*, in "Harvard Law Review", 90, 1977, pp. 837-893.
- MCCARTY T., *Some Requirements for a Computer-Based Legal Consultant*, Report LRP-TR-8, Laboratory for Computer Science Research, Rutgers University, 1980.
- MCCARTY T., *The TAXMAN Project: Towards a Cognitive Theory of Legal Argument*, in NIBLETT B. (a cura di), *Computer Science and Law: An Advanced Course*, 1980, pp. 23-43.
- MCCARTY T., *A Computational Theory of Eisner v. Macomber*, in CIAMPI C. (a cura di) *Artificial Intelligence and Legal Information Systems*, 1982, pp. 329-355.
- MCCARTY T., *A Language for Legal Discourse. I Basic Features*, relazione presentata al congresso internazionale *Artificial Intelligence and Legal Reasoning* (Vancouver, giugno 1989).
- MCCARTY T., *Clausal Intuitionistic Logic. I. Fixed Point Semantics*, in "Journal of Logic Programming", 5 (1), 1988, pp. 1-31
- MCCARTY T., *Clausal Intuitionistic Logic. II. Tableau Proof Procedures*, in "Journal of Logic Programming", 5 (2), 1988, pp. 93-132
- MCCARTY T., *Intelligent Legal Information Systems: an Update*, in *Law and Computers*, 5, 1987, pp. 196-202 (Law and Computers Association of Japan) (pubblicato anche in FIEDLER H., HAFT F., TRAUNMÜLLER R. (a cura di), *Expert Systems in Law, Impacts on Legal Theory and Computer Law*, Attempto Verlag, Tübingen, pp. 15-25).
- MCCARTY T., *Intelligent Legal Information Systems: Problems and Prospects*, in "Routgers Computer & Technology Law Journal", 9, 1983, pp. 265-294 (pubblicato anche in CAMPBELL (a cura di), *Data Processing and the Law*, Sweet and Maxwell, 1984, pp. 125-151).
- MCCARTY T., *Permission and Obligation: an Informal Intruduction*, in MARTINO A.A., SOCCI NATALI F., *Automated Analysis of Legal Texts*, 1986, pp. 307-337.
- MCCARTY T., SHIDARAN N. S., *The Representation of Conceptual Structures in TAXMAN II, Part One: Logical Templates*, LPR, TR, 4, Laboratory for Computer Science Research, Rutgers University, 1980.

- MCCARTY T., SHIDARAN N.S., *The Representation of Evolving Systems of Legal Concepts*, Technical Report LPR-TR-13, Computer Science Department, Rutgers University, 1981.
- MEHL L., *Automation in the legal world. From the Machine Processing of Legal Information to the 'Law machine'*, in *Mechanisation of Thought Processes*. National Physical Laboratory Symposium No 10, novembre 1958., Her Majesty's Stationery Office, London, 1959, pp. 757-787.
- MELDMAN J.A., *A Preliminary Study in Computer-aided Analysy*, MIT Report MAC TR-157, M.I.T., Cambridge (Massachusetts), 1975.
- MELDMAN J.A., *A Structural Model for Computer Aided Legal Analysis*, in "Rutgers Journal of Computers and the Law", 1977, pp. 27-71
- MENDELSON E., *Introduction to Mathematical Logic*, D. Van Nostrand, Princeton (NJ), 1964 (trad. it. MENDELSON E., *Introduzione alla logica matematica*, Boringhieri, Torino, 1972).
- MERLO B., MICHELI G., *Il condono edilizio computerizzato*, Pirola, Milano, 1985.
- MICHAELSEN R.H., *An Expert System for Federal Tax Planning*, in "Expert Systems", 1, 1984, p. 240 ss.
- MINSKI M., *A Framework for the Representation of Knowledge*, in WINSTON P.H. (a cura di), *The Psychology of Computer Vision*, McGraw-Hill, New York, 1975, pp. 211-280 (già pubblicata come *Artificial Intelligence Memo 306*, MIT AI Lab, 1974)
- MONTAGUE R., *Formal Philosophy. Selected Papers of Richard Montague*, THOMASON R.H. (a cura di), Yale University Press, New York, 1974.
- MOORE G.E., *Principia Ethica*, Cambridge, 1903 (trad. it. *Principia Ethica*, Bompiani, Milano, 1964).
- NAGEL STUARD S., *Microcomputers and Judicial Predictions*, in MARTINO A.A., SOCCI NATALI F., *Automated Analysis of Legal Texts*, 1986, pp. 681-702.
- NEWELL A., SHAW J.C., SIMON H.A., *Empirical Exploration with the Logic Theory Machine: a Case Study in Heuristics*, in FEIGENBAUM E.A., FELDMAN J. (a cura di) *Computers and Thought*, 1963, pp. 109-133.
- NEWELL A., SIMON H.A., *Computer Science as Empirical Enquiry: Symbols and Search*, in "Communication of the ACM", Vol 19, n°3, 1976, pp. 113-126.

- NEWELL A., SIMON H.A., *GPS, a Program that Simulates Human Thought*, in FEIGENBAUM E.A., FELDMAN J. (a cura di) *Computers and Thought*, 1963, pp. 279-293.
- NEWELL A., SIMON H.A., *Human Problem Solving*, Prentice-Hall Englewood Cliffs (NJ), 1972.
- NIBLETT B. (a cura di), *Computer Science and Law: An advanced Course*, Cambridge University Press, Cambridge, 1979.
- NOVELLI V., GIANNANTONIO E., *Manuale per la ricerca elettronica dei documenti giuridici*, Giuffr , Milano, 1982.
- ODDY R., ROBERTSON S., VAN RIJSBERGEN C., WILLIAMS P., *Information Retrieval Research*, Butterworths, London, 1981.
- OELINGER T. (a cura di), *Gesetzgebung und Computer*, Schweizer Verlag, M nchen, 1981.
- PATTARO E., CASADEI G., SARTOR G., *An Expert System in Environmental Law (IRI Project)*, relazione presentata al Congresso internazionale *Expert Systems in Law* (Bologna, maggio 1989).
- PATTARO E., *Introduzione al corso di filosofia del diritto. Volume primo*, Clueb, Bologna, 1986.
- PATTARO E., *Introduzione al corso di filosofia del diritto. Volume secondo*, Clueb, Bologna, 1987.
- PATTARO E., *Lineamenti per una teoria del diritto*, Clueb, Bologna, 1985.
- PATTARO E., *Models of Reason, Types of Principles and Reasoning. Historical Comments and Theoretical Outlines*, in "Ratio juris", 2, 1988, pp. 109-122.
- PECZENIK A., *Legal reasoning as a special case of moral reasoning*, in "Ratio juris", 2, 1988, pp. 123-136.
- PERELMAN C., *Justice et raison*, Editions de l'universit  de Bruxelles, Bruxelles, 1972 (prima edizione 1963).
- PERELMAN C., *Logique juridique. Nouvelle rh torique*, Dalloz, Paris, 1979 (2a ed.).
- PERELMAN C., OLBRECHTS-TYTECA L., *La nouvelle rethorique. Trait  de l'argumentation*, Presses Universitaires de France, Paris 1958 (2a ed. Brouxelles 1970).
- POOLE D., *A Logical Framework for Default Reasoning*, in "Artificial Intelligence", 36, 1988, pp. 27-47.

- POPP W.G., SCHLINK B., *JUDITH: a Computer Programme to Advise Lawyers in Reasoning a Case*, in "Jurimetrics Journal", 15, 1975, pp. 303-314
- POPPER K.R., *On the sources of knowledge and ignorance*, in *Conjectures and Refutations*, Routledge and Kegan Paul, London, 1963.
- PUTNAM H., *Formalizzazzione*, in *Enciclopedia Einaudi*, 6, 1979, pp. 324-341.
- PUTNAM H., *The Meaning of Meaning*, in GUNDERSON K. (a cura di) *Language, Mind and Knowledge*, Minneapolis, University of Minnesota Press, 1975, pp. 131-193.
- QUINE V. O. , *Methods of Logic*, Routledge and Kegan Paul, London, 1974 (prima ed. 1952).
- REISINGER L., *Approach to a Theory of "Soft Algorithms"*, in MARTINO A.A., SOCCI NATALI F., *Automated Analysis of Legal Texts*, 1985, pp. 761-772.
- REITER, *On Closed World Data Bases*, in GALLAIRE, MINKER (a cura di), *Logic and Data Bases*, 1978, pp. 55-76.
- RICH E., *Artificial Intelligence*, in SHAPIRO S.C., ECKROTH D., *Encyclopedia of Artificial Intelligence*, pp. 9-16.
- RICH E., *Intelligenza artificiale*, McGraw, Hill, Milano, 1986.
- RISSLAND E.L., *Argument Moves and Hypotetycals*, in WALTER C. (a cura di), *Computing Power and Legal Reasoning*, 1985, pp. 129-143.
- RISSLAND E.L., ASHLEY K.D., *A Case-based System for Trade Secrets Law*, in *Proceedings of the First International Conference on Artificial Intelligence and Law* (Boston, maggio 1987), 1987, pp. 60-66.
- RISSLAND E.L., SOLOWAY E.M., *Overview of an Example Generation System*, in *Proceedings of the National Conference on Artificial Intelligence (AAAI-80)* (Stanford (California), Agosto 1980), 1980.
- RISSLAND E.L., VALCARCE E.M., ASHLEY K.D., *Explaining and Arguing with Examples*, in *Proceedings of the National Conference on Artificial Intelligence (AAAI-84)*, Austin (Texas), 1984, pp. 288-294.
- ROBINSON J.A., *A Machine-Oriented Logic Based on the Resolution Principle*, in JACM, 12, 1965, pp. 23-41

- ROBINSON J.A., *Logic: Form and Function . The Mechanization of Deductive Reasoning*, North Holland Elsevier, New York, 1979.
- RÖDIG J., *Schriften zur juristischen Logik* (a cura di BUND E., SCHMIEDEL G., THIELER-MEVISSSEN G.) , Springer Verlag, Berlin, 1980
- RÖDIG J., *Axiomatisierbarkeit juristischer Systeme*, in RÖDIG J., *Schriften zur Juristischen Logik*, 1980, pp. 40-90 (già in KAUFMANN A. (a cura di) *Münchner Ringvorlesung EDV und Recht. Möglichkeiten und Probleme*, in *EDV und Recht*, 6, 1973, pp. 49-90)
- ROSS A.,  $T_0$ - $t_0$ , in "Scandinavian Studies in Law", I, 1957, pp. 139-153 (pubblicato anche in "Harvard Law Review", 70, 1956-1957, pp. 812-825; trad it. in SCARPELLI U. (a cura di), *Diritto e analisi del linguaggio*, Edizioni di comunità, Milano, 1976).
- ROSS A., *On Law and Justice*, Steven and Sons, London 1958 (trad. it. *Diritto e giustizia*, Einaudi, Torino, 1965).
- SACCO R., *La nozione del contratto*, in *Trattato di diritto privato*, 10 *Obbligazioni e contratti*, Tomo secondo, UTET, Torino, 1982, pp. 5-15.
- SANCHEZ-MAZAS M., *Algebraic and Arithmetical Translations on Normative Systems and Applications in Legal Informatics*, in MARTINO A.A. (a cura di) *Deontic Logic, Computational Linguistics and Legal Information Systems*, 1986, pp. 169-201.
- SANCHEZ-MAZAS M., *Un modello aritmetico come sistema esperto per il confronto e l'armonizzazione di legislazione*, relazione presentata al Congresso internazionale *Expert Systems in Law*, (Bologna, maggio 1989).
- SANTORO PASSARELLI F., *Dottrine generali del diritto civile*, Jovene, Napoli, 1966.
- SARTOR G., *Alcune osservazioni sull'applicabilità della logica classica al diritto*, in "Informatica e diritto", 3, 1987, pp. 65-90.
- SARTOR G., *Applicazioni dell'intelligenza artificiale al diritto: il progetto IRI (Irnerio Regioni Italiane)*, Relazione presentata al 41 congresso internazionale sul tema *Informatica e regolamentazioni giuridiche* (Roma, maggio 1988).
- SARTOR G., *Le reti di Petri Un linguaggio formale applicabile al diritto?*, in "Sociologia del diritto", 1986, pp. 109-118.
- SARTOR G., *The Irinorm System: Drafting of Normalized Texts and Generation of Knowledge Bases in Predicate Logic*, relazione

presentata al Congresso internazionale Expert Systems in Law, (Bologna, maggio 1989).

- SCARPELLI U., *Contributo alla semantica del linguaggio normativo*, Memoria dell'Accademia delle scienze di Torino, serie 3 tomo 5 parte II n. 1, Torino, 1959.
- SCARPELLI U. (a cura di), *Diritto e analisi del linguaggio*, Edizioni di comunità, Milano, 1976.
- SCARPELLI U., *Cosa é il positivismo giuridico*, Edizioni di comunità, Milano, 1965.
- SCHANK R.C., *Conceptual Information Processing*, Elsevier, New York, 1975.
- SCHANK R.C., ABELSON R.P., *Scripts, Plans, Goals and Understanding*, Erlbaum, Hillsdale (NJ), 1977.
- SCHAUSS M., *Prototyp eines Informationssystem zur Hilfe bei der juristischen Entscheidungsfindung*, in ERDMANN U., FIEDLER H., HAFT F., TRAUNMÜLLER R. (cura di), *Computergestützte juristischer Expertensysteme*, 1986, pp. 153-164.
- SCHLOBOHM A., *TA-- A PROLOG Program which Analyses Income Tax Issues unter Section 318 (A) on the Internal Revenue Code*, in WALTER C. (a cura di), *Computing Power and Legal Reasoning*, 1985, pp. 765-815
- SEARLE J.R., *Minds, brains and programs*, in *Behav. Brain Sci.* 3, 1980, pp. 417-457.
- SEARLE J.R., *Minds, Brains and Science*, The 1984 Reith lectures, British Broadcasting Corporation, Pitnam Press, Bath, 1984, pp. 102
- SENECA L. A., *Ad Lucilium epistularum moralium libri* (trad.it. *Lettere a Lucilio*, Rizzoli, Milano, 1987).
- SERGOT M.J. *A Query-the-User Facility for Logic Programming*, in DEGANI P., SANDEWALL E. (a cura di), *Integrated Interactive Computer Systems*, North-Holland, Amsterdam, 1983.
- SERGOT M.J., CORY H.T., HAMMOND P., KOWALSKI R.A., KRIWACZEK F., SADRI F., *Formalisation of the British Nationality Act*, in ARNOLD C. (a cura di) *Yearbook of Law, Computers and Technology: Volume 2*, 1986.
- SERGOT M.J., HAMMOND P., *A PROLOG Shell for Logic Based Expert Systems*, in *Expert systems 83: Proceedings of the 3rd Technical Conference of the British Computer Society Specialist Group on*

- Expert Systems* (Cambridge, Dicembre 1983), British Computer Society, 1983, pp. 94-104.
- SERGOT M.J., *Logic Programming and its Applications in Law: a Brief Tutorial Introduction*, relazione presentata 2nd International Conference on Law and Technology, (University of Huston Law Centre, giugno 1985).
- SERGOT M.J., *Programming Law: LEGOL as a Logic Programming Language*, Department of Computing and Control, London, 1980.
- SERGOT M.J., *Prospects for Representing the Law as Logic Programs*, in CLARK K.L., TARNLUND S.A. (a cura di), *Logic Programming*, 1982, pp. 33-42.
- SERGOT M.J., SADRI F., KOWALSKI R.A., KRIWACZEK F., HAMMOND P., CORY H.T., *British Nationality Act as a Logic Programme*, in "Communication of the Association for Computing Machinery", 29,5, 1986, pp. 370-386.
- SERGOT M.J., *The Representation of Law in Computer Programs: A Survey and Comparison of Past and Current Projects*, 1987 (giugno-luglio).
- SHAPIRO S.C., ECKROTH D., *Encyclopedia of Artificial Intelligence*, John Wiley & Sons, New York, 1987.
- SHAPIRO S.C., *Processing, Bottom-up and Top-down*, in SHAPIRO S.C., ECKROTH D., *Encyclopedia of Artificial Intelligence*, 1987, pp. 779-785
- SHIDARAN N.S., *Representing Knowledge in AIDMS*, in "Informatica e diritto", 7, 1981, pp. 201-202.
- SIMON H.A., *The Sciences of Artificial*, The M.I.T. Press, Cambridge (Massachusetts), 1969 (trad it. *Le scienze dell'artificiale*, ISEDI, Milano, 1973).
- SIMON H.A., *The New Science of Management Decisions*, Prentice-Hall, Englewood Cliffs (New Jersey), 1977 (trad. it, *Informatica, direzione aziendale e organizzazione del lavoro*, Franco Angeli, Milano, 1979).
- SMITH E.E., MEDIN D.L., *Categories and Concepts*, Harvard University Press, Cambridge (Massachusetts), 1981.
- SOWA J.F., *A Prolog to Prolog*, in WALKER A., MCCORD M., SOWA J.F., WILSON W.G., *Expert Systems and Prolog*, Addison, Wesley, Reading (Massachusetts), 1987, pp. 25-108.



- SOWA J.F., *Conceptual Structures: Information Processing in Mind and Machine*, Addison, Wesley, Reading (Massachusetts), 1984.
- SOWA J.F., *Semantic Networks*, in SHAPIRO S.C., ECKROTH D., *Encyclopedia of artificial intelligence*, John Wiley & Sons, New York, 1987, pp. 1011-1024.
- SPROWL J.A., *Assembling Standardized Legal Documents in a Semiautomatic Fashion with a Rule-Based Expert System*, relazione presentata al Congresso internazionale *Expert Systems in Law*, (Bologna, maggio 1989).
- SPROWL J.A., *Automated Assembly of Legal Documents*, in NIBLETT B. (a cura di), *Computer Science and Law*, 1980, pp. 195-205.
- SPROWL J.A., *Automating the Legal Reasoning Process: A Computer that Uses Regulations and Statutes to Draft Legal Documents*, in "American Bar Foundation Research Journal", 1979, pp. 1-81.
- STAMPER R.K., *LEGOL: Modelling Legal Rules by Computer*, in NIBLETT (a cura di), *Computer Science and Law*, 1980, pp. 45-71.
- STAMPER R.K., *A Non-classical Logic Based on the Structures of Behaviour*, in MARTINO A.A., SOCCI NATALI F., *Automated Analysis of Legal Texts*, 1985, pp. 115-139.
- STAMPER R.K., BACKHAUS J., ALTHAUSE K., *Expert Systems: Layers Beware*, relazione presentata al 4<sup>o</sup> Congresso internazionale sul tema "Informatica e regolamentazioni giuridiche" (Roma, maggio 1988).
- STAMPER R.K., *Information: Mystical Fluid of Subject for Scientific Enquiry*, in "The Computer Journal", vol. 28, 2, 1985.
- STAMPER R.K., TAGG C., MASON P., COOK S., MARKS J., *Developing the LEGOL Grammar*, in CIAMPI (a cura di) *Artificial Intelligence and Legal Information Systems*, 1982, pp. 357-287.
- STAMPER R.K., *The automation of legal reasoning: Problems and prospects*, LEGOL Project Report L39, LSE Systems Research Group, London School of Economics and Political Science, London, 1976
- STAMPER R.K., *The Role of Semantics in Legal Expert Systems and Legal Reasoning*, relazione presentata al Congresso internazionale *Expert Systems in Law* (Bologna, maggio 1989).
- STEINMÜLLER W., 1981, *Informationstechnologie Folgen Informationstechnologie und Informationsrecht III*, in "Nachrichten für Dokumentation", pp. 241-255.

- STERLING L., SHAPIRO E., *The Art of Prolog*, MIT Press, Cambridge (Massachusetts), 1987.
- STRIPINIS, *Fiscal Legislation Planner*, in MARTINO A.A., SOCCI NATALI F., *Automated Analysis of Legal Texts*, 1986, pp. 821-830.
- SUSSKIND R.E., *Expert Systems in Law and the Data Protection Adviser*, in "Oxford Journal of Legal Studies", 7, 1987, pp. 145
- SUSSKIND R.E., *Expert Systems in Law. A Jurisprudential Inquiry*, Claredon Press, Oxford, 1987.
- SVOBODA W.R., *Models in Planning Legislation*, in MARTINO A.A., SOCCI NATALI F., *Automated Analysis of Legal Texts*, 1986, pp. 831-840.
- TAMMELO I., *Modern Logic in the Service of Law*, Springer, Berlin, 1978.
- TARELLO G., *Formalismo*, in *Novissimo digesto italiano*, VII, 1961, 572-580
- TARELLO G., *L'interpretazione della legge*, Giuffrè, Milano, 1980, pp. 419.
- TARELLO G., *Realismo giuridico*, in *Novissimo digesto italiano*, XIV, 1968, pp. 923-933
- TARELLO G., *Storia della cultura giuridica moderna I. Assolutismo e codificazione del diritto*, Il Mulino, Bologna, 1976.
- TARSKI, *Der Wahrheitsbegriff in den formalisierten Sprachen*, in "Studia Philosophica", 1, 1936, pp. 261-405 (trad. it. *Il concetto di verità nei linguaggi formalizzati*, in RIVETTI, BARBO', *L'antinomia del mentitore nel pensiero contemporaneo da Peirce a Tarski*, 1961)
- THOMASSET C., *Expert System in Québec Housing Law: from HOME-Expert I to HOME-Expert II*, Relazione presentata al Congresso internazionale *Expert Systems in Law* (Bologna, maggio 1989).
- TUNER R., *Logics for Artificial Intelligence*, Ellis Horwood, Chichester (England), 1984.
- ULLMAN J.D., *Principles of Databases Systems*, Computer Science Press, Potomac (Maryland), 1980.
- VAN DALEN D., *Logic and Structures*, Springer, Berlin, 1980.
- VIEHWEG T., *Topik und Jurisprudenz. Ein Beitrag zur rechtswissenschaftlichen Grundlagenforschung*, C.H. Beck'sche Verlagsbuchhandlung, München, 1965 (prima ed. 1954).

- WALKER A., *Knowledge systems: Principles and Practice*, in "IBM Journal of Research and Development", 1986, pp. 2-14.
- WALKER A., *Knowledge systems: Principles and Practice*, in WALKER A., MCCORD M., SOWA J.F., WILSON W.G., *Expert Systems and Prolog*, 1987, pp. 1-24.
- WALKER A., MCCORD M., SOWA J.F., WILSON W.G., *Expert Systems and Prolog*, Addison-Wesley, Reading (Massachusetts), 1987, pp. 475
- WALTER C. (a cura di), *Computing Power and Legal Reasoning*, West Publishing Company, St. Paul (Minnesota), 1985.
- WATERMAN D.A., *New Research Tools to Watch for*, in "California Law Review", 5(3), 1985, pp. 19-21).
- WATERMAN D.A., PETERSON M.A., *Evaluating Civil Claims: an Expert Systems Approach*, in "Expert systems", 1, 1984, pp. 65-76.
- WATERMAN D.A., PETERSON M.A., *Rule-based models of legal expertise*, in *Proceedings of the First Annual Conference on Artificial Intelligence* (Stanford univ., Aug.. 1980), 1981, pp. 272-275
- WATERMAN D.A., PETERSON M.A., *Expert System for Legal Decision Making*, in "Expert systems", 3, 1986, pp. 212-226
- WATERMAN D.A., PETERSON M.A., *Models of Legal Decisionmaking*, in KLAHR P., WATERMAN D.A., *Expert Systems. Techniques, Tools and Applications*, pp. 135-185.
- WEGNER P., *Research Paradigms in Computer Science*, in *Proceedings of the International Conference on Reliable Software* (Los Angeles, aprile 1975).
- WEINBERGER C., WEINBERGER O., *Logik, Hermeneutik und Semantik*, Beck, München, 1979.
- WEISS S., KULIKOWSKI C., AMAREL S., SAFIR A., *A Model-Based Method form Computer-Aided Medical Decision-Making*, in "Artificial Intelligence", 11, 1978, pp. 145-172.
- WEIZENBAUM J., *Computer Power and Human Reason. From Judgement to Calculation*, Freeman, San Francisco (California), 1977.
- WINOGRAD T, FLORES C.F., *Understanding Computer and Cognition*, Ablex Publishing Co., Norwood (New Jersey), 1986.
- WINSTON P.H., *The Psychology of Computer Vision*, McGraw Hill, New York, 1975.

- WIRTH N., *Algorithms + Data Structures = Programs*, Prentice Hall, Englewood Cliffs (New Jersey).
- WITTGENSTEIN L., *Tractatus Logico-philosophicus e Quaderni 1916*, Einaudi, Torino, 1968 (1aed. 1964), pp. 270.
- WITTGENSTEIN L., *Philosophische Untersuchungen*, in *Werkausgabe Band 1*, 1984 (1a ed 1952), pp. 225-618.
- WITTGENSTEIN L., *Werkausgabe Band 1. Tractatus logico-philosophicus. Tagebucher 1914-1916. Philosophische Untersuchungen*, Suhrkamp, Frankfurt, 1984, pp. 620.
- WRIGHT VON, *Bedingungsnormen. Ein Prüfstein für die Logik der Normen*, in KRAVIETZ W., SCHELSKI H., WINKLER G., SCHRAMM A. (a cura di), *Theorie der Normen. Festgabe für Ota Weinberger zum 65 Geburtstag*, 1984, pp. 447-456.
- WRIGHT VON H., *Deontic Logic*, in "Mind", 60, 1951, pp. 1-15 (trad. it. W., *Logica deontica*, appendice di DI BERNARDO G., *Logica, norme, azione*, 1969, pp. 119-148).
- WRIGHT VON H., *Problemi e prospettive della logica deontica. Una panoramica*, 1969, pp. 383-405.
- WROBLEWSKI J., *Facts in Law*, in *Meaning and Truth in Judicial Decision*, 1983, pp. 104-126 (già in ARSP, 1973, p. 161 ss).
- WROBLEWSKI J., *Justification of Legal Decisions*, in *Meaning and Truth in Judicial Decision*, 1983, pp. 49-70 (già in "Logique et Analyse", 1969, p. 3 ss)
- WROBLEWSKI J., *Legal Reasoning in Legal Interpretation*, in *Meaning and Truth in Judicial Decision*, 1983, pp. 71-103 (già in "Logique et analyse", p. 3 ss)
- WROBLEWSKI J., *Semantic Basis of the Theory of Legal Interpretation*, in *Meaning and Truth in Judicial Decision*, 1983, pp. 22-48 (già in "Logique et Analyse", 1963, p. 397 ss)
- WROBLEWSKI J., *Meaning and Truth in Judicial Decision*, A-THIETO Oy, Helsinki, 1983
- ZADEH L.A., *Fuzzy Logic and its Application to Approximate Reasoning*, in *Information Processing 74*, North Holland, Amsterdam, 1974, pp. 591-594.

## INDICE DEL VOLUME

<b>Introduzione</b>	<b>6</b>
---------------------	----------

### PARTE I

### INFORMATICA E AUTOMAZIONE DEL LAVORO INTELLETTUALE

<b>CAP. 1. L'automazione del lavoro intellettuale: dalla macchina calcolatrice ai sistemi di conoscenza</b>	<b>7</b>
---	----------

1. Le operazioni di calcolo numerico	9
2. La gestione di dati	10
3. I comportamenti intelligenti	15
3.1. L'intelligenza artificiale	15
3.2. I linguaggi dell'intelligenza artificiale	20
3.3. Intelligenza artificiale, spazio di stati e sistemi assiomatici formali	22
3.4. I limiti del paradigma del sistema assiomatico formale	26
3.4.1 Il problema dell'esplosione combinatoria	29
3.4.2 Il problema della formalizzazione	30

<b>CAP. 2. L'automazione del lavoro giuridico: dalla giurimetria all'intelligenza artificiale</b>	<b>33</b>
---	-----------

1. Operazioni di calcolo	34
2. La gestione di dati	35
2.1. L'applicazione automatica di norme giuridiche	35
2.1.1. Applicazioni gestionali	35
2.1.2. Sistemi per consulenze giuridiche	36
2.1.3. Programmazione tradizionale e applicazione del diritto.	38
2.1.4. La programmazione strutturata come modello del diritto	41
2.2. L'automazione d'ufficio come supporto dell'attività del giurista	43

2.3. Le basi di dati giuridiche	44
2.4. Conclusione: Limiti dell'uso degli strumenti informatici tradizionali nel diritto	47
3. I comportamenti intelligenti	48
3.1. Il problema dell'esplosione combinatoria nel diritto	49
3.2. Il problema della formalizzazione nel diritto (rinvio)	50

### **CAP. 3. I sistemi basati sulla conoscenza 54**

1. I sistemi di conoscenza	54
2. I sistemi esperti	56
3. Sviluppo della tecnologia dei sistemi di conoscenza	57
3.1. La tesi	57
3.2. L'antitesi	58
3.3. La sintesi	58
4. La programmazione logica (cenni)	59

### **CAP. 4. I sistemi basati sulla conoscenza giuridica 63**

1. 1. I predecessori (cenni)	66
2. I principali progetti degli anni 70	68
2.1. Il progetto TAXMAN	68
2.2. Il progetto CLLIPS	71
2.3. Il sistema JUDITH	72
2.4. Il progetto di Meldman	72
2.5. Il progetto LEGOL	74
2.6. Logica e diritto nell'opera di L.A. Allen	76
2.7. Il sistema LIRS	79
3. I principali progetti degli anni 80	80
3.1. La programmazione logica: il progetto PROLOG	80
3.2. Il trattamento del linguaggio naturale: il progetto LEX	84
3.3. Gli aspetti strategici: i sistemi LSD e SAL	86
3.4. L'uso dei precedenti: il progetto HYPO	88
3.5. Altri progetti	90
4. Conclusione	97
4.1. Contenuti e prestazioni dei sistemi basati sulla conoscenza giuridica	97
4.2. Gli usi dei sistemi basati sulla conoscenza giuridica	104
4.2.1. La soluzione di questioni di diritto	104
4.2.2. Una funzione previsionale o strategica basata (anche) sui fatti	106
4.2.3. La redazione di documenti giuridici	106

4.2.4. La ricerca di informazioni giuridiche	107
4.2.5. L'ausilio nella legislazione	108
4.2.6. L'apprendimento del diritto	109

## PARTE II

### LA RAPPRESENTAZIONE DELLA CONOSCENZA

#### **CAP. 5. Rappresentazione della conoscenza e intelligenza artificiale** 111

1. Il problema della formalizzazione	113
1.3 Il concetto di formalizzazione	114
1.4 Necessità di una formalizzazione	116
1.2 Conoscenze pubbliche e private	117
2. Rappresentazione della conoscenza e significati	119
2.1 Rappresentazione della conoscenza, teorie del significato e strumenti informatici	120
2.2 L'ereditarietà delle proprietà	121
2.3 Concezioni del significato e tecniche definitorie	122
2.4 Problemi del linguaggio giuridico (cenni)	124
2.5 Conclusione: teorie del significato e strumenti informatici	126

#### **CAP. 6 Rappresentazione e formalizzazione della conoscenza giuridica** 128

1. L'assiomatizzazione del diritto	128
2. Scienze naturali e scienze giuridiche	132
3. Ragionamento giuridico e metodo assiomatico	136
3.1. Giustificazione interna e giustificazione esterna della decisione giuridica	138
3.2. Modelli di razionalità e di giustificazione nei sistemi informatico-giuridici	138
3.3. Relatività delle premesse prescrittive e controllo dei sistemi basati sulla conoscenza giuridica	142
3.4. Conclusione. Ragionamento giuridico e razionalità dei sistemi informatico-giuridici intelligenti	144
4. Interpretazione del diritto e rappresentazione della conoscenza	145
4.1. Il concetto di interpretazione	145

4.2. Interpretazione e testi	148
4.3. Interpretazione e competenze dell'ingegnere della conoscenza giuridica	151
4.4. Il problema dell'interpretazione del fatto	152
5. Conoscenze giuridiche pubbliche e private	155
6. Conoscenze e posizioni giuridiche	157
7. Conoscenze profonde e superficiali	157
8. Il contributo della filosofia e della teoria del diritto	160

## **CAP. 7. Metodi e linguaggi per la rappresentazione della conoscenza nell'intelligenza artificiale**

163

1. La logica	163
1.1. La logica enunciativa	164
1.2. La logica dei predicati del primo ordine	168
1.3. Altre logiche	172
1.3.1. Le logiche modali	172
1.3.2. Le logiche deontiche	175
2. La programmazione logica	177
2.1. Il PROLOG: la rappresentazione della conoscenza	180
2.2. I quesiti	181
2.3. Il motore inferenziale.	184
3. Le regole di produzione	189
4. Le reti semantiche	192
5. I frames	197

## **CAP. 8. Metodi e linguaggi per la rappresentazione della conoscenza giuridica**

200

1. La logica nel diritto (cenni)	200
1.1. La logica enunciativa. La normalizzazione di L.E. Allen	203
1.1.1 Primo esempio : la normalizzazione della normativa sulla capacità di contrarre matrimonio	206
a) Il requisito dell'età	206
b) Le condizioni del potere di contrarre matrimonio	207
1.1.2 Secondo esempio, la normalizzazione di un articolo della costituzione italiana	207
a) Testo originale	207
b) Formula logica	207
c) Testo normalizzato	208
d) Schema del testo normalizzato	208
Diagramma a freccia	209



1.1.3 Il significato della normalizzazione	209
1.2. La logica predicativa	212
1.2.1 Formalizzazione di un principio generale del diritto	212
a) Principio	212
b) Glossario:	213
c) Formalizzazione nella logica predicativa	213
1.2.2 Formalizzazione di un'argomentazione giudiziaria	213
a) Fatti della causa:	213
b) Testo dell'argomento della difesa:	213
c) Struttura logica dell'argomento:	214
d) Glossario:	214
e) Premesse dell'argomento:	214
f) Conclusione	215
g) Deduzione	215
2. La programmazione logica	216
2.1. La programmazione logica in senso stretto	216
2.1.1 La formalizzazione del British Nationality Act2	17
a) Testo originale:	218
b) Rappresentazione in clausole di Horn.	218
2.2. La teoria generale del discorso	220
2.2.1 Conoscenze del senso comune	223
2.2.2 Conoscenze giuridiche.	223
2.3. Il linguaggio per il discorso giuridico di McCarty	225
2.3.1 La sintassi delle formule atomiche	226
2.3.2 La sintassi delle formule molecolari	228
a) Le clausole di Horn	228
b) Negazioni e implicazioni annidate	228
c) Il ragionamento per difetto	229
d) Prototipi e deformazioni	230
2.3.3 Le modalità	231
a) Il tempo	231
b) Eventi ed Azioni	232
c) I concetti deontici	232
d) Conclusione	233
3. Le regole di produzione	234
3.1. Le norme ipotetiche	234
3.2. Il diritto è riducibile ad un insieme di obblighi condizionati?	237
3.3. Il diritto è riducibile ad un insieme di regole di produzione?	240
3.3.1 Esempio di regole di produzione giuridiche	240
3.3.2 Regole di produzione e programmazione logica	241
4. Le reti semantiche	242
5. I frames	245
6. Problemi e limiti della formalizzazione del diritto nella programmazione logica	248

6.1. Regole con conclusioni disgiuntive	249
6.2. La negazione	250
6.2.1 Conclusioni negative e negazione come fallimento	251
6.2.2 Ipotesi del mondo chiuso e diritto	253
a) Ipotesi del mondo chiuso e questioni di diritto	253
b) Ipotesi del mondo chiuso e questioni di fatto	255
c) Conclusione. Negazione nella programmazione logica e nel diritto.	260
6.3. I condizionali soggettivi	261
6.4. La considerazione dei tipi	262
6.5. I concetti intensionali	264
6.6. L'interpretazione del fatto	266
6.7. Conclusione	266

## **CAP. 9. Intelligenza artificiale, logica e rappresentazione formale della conoscenza giuridica 268**

1. Intelligenza artificiale e logica	268
2. Due critiche all'uso di modelli logici nell'informatica giuridica	273
2.1. La critica di Leith	273
2.2. La critica di Stamper	274
2.2.1 La metafora del condotto	275
2.2.2 La metafora dell'elaborazione di dati	276
2.2.3 La metafora insiemistica della realtà.	277
2.2.4 La metafora della corrispondenza per il significato e la verità	277
2.2.5 La metafora della mente come elaborazione di informazioni.	278
3. Limiti dell'intelligenza artificiale giuridica	278
4. Limiti dell'approccio logico nel diritto	281

## **APPENDICE. Il programma IRINORM 285**

1. La struttura del sistema	285
2. Principi ispiratori	286
2.1 La tecnica della normalizzazione	286
2.2 L'uso della logica per esprimere contenuti giuridici	291
3. L'ambiente di sviluppo	293
3.1 L'inserimento e la modifica della conoscenza	293
4. L'ambiente di consultazione	301
4.1 L'interrogazione del sistema	301

4.2 Il controllo della deduzione e della base di conoscenza	303
---	-----

<b>BIBLIOGRAFIA. Elenco delle opere citate</b>	<b>305</b>
--	------------









